



**TUGAS AKHIR - KS141501**

**RANCANG BANGUN APLIKASI KOPERASI SIMPAN  
PINJAM DENGAN METODE VIEWPOINT ORIENTED  
REQUIREMENT DEFINITION.**

**(Studi kasus: Koperasi Karyawan PT Wonosari  
Jaya)**

**Alvisha Farrasita Istifani  
NRP 0521134 0000 057**

**Dosen Pembimbing 1:  
Sholiq, S.T ,M.Kom, M.SA**

**DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018**



**FINAL PROJECT - KS141501**

**DESIGN BUILDING APPLICATION OF SAVE LOANS BY THE  
VIEWPOINT ORIENTED REQUIREMENT DEFINITION  
METHOD.**

**(Case study: Employee Cooperative PT Wonosari Jaya)**

**Alvisha Farrasita Istifani  
NRP 0521134 0000 057**

**Supervisor :  
Sholiq, S.T ,M.Kom, M.SA**

**DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018**

**TUGAS AKHIR - KS 141501**

**RANCANG BANGUN APLIKASI KOPERASI  
SIMPAN PINJAM DENGAN METODE  
VIEWPOINT ORIENTED REQUIREMENT  
DEFINITION.  
(Studi kasus: Koperasi Karyawan PT  
Wonosari Jaya)**

**Alvisha Farrasita Istifani  
NRP 0521134 0000 057**

**Dosen Pembimbing 1:  
Sholiq, S.T ,M.Kom, M.SA**

**DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018**



**ITS**  
Institut  
Teknologi  
Sepuluh Nopember

**FINAL PROJECT - KS 141501**

# **DESIGN BUILDING APPLICATION OF SAVE LOANS BY THE VIEWPOINT ORIENTED REQUIREMENT DEFINITION METHOD.**

**(Case study: Employee Cooperative PT  
Wonosari Jaya)**

**Alvisha Farrasita Istifani  
NRP 0521134 0000 057**

**Supervisor :  
Sholiq, S.T ,M.Kom, M.SA**

**DEPARTEMEN SISTEM INFORMASI  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember  
Surabaya 2018**

## LEMBAR PENGESAHAN

**RANCANG BANGUN APLIKASI KOPERASI SIMPAN  
PINJAM DENGAN METODE VIEWPOINT ORIENTED  
REQUIREMENT DEFINITION. (Studi kasus: Koperasi  
Karyawan PT Wonosari Jaya)**

### **TUGAS AKHIR**

Disusun untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Departemen Sistem Informasi  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember

Oleh:

**Alvisha Farrasita Istifani**  
0521134 0000 057

Surabaya, 13 Desember 2017



**KEPALA  
DEPARTEMEN SISTEM INFORMASI**

**Dr. Ir. Kris Tjahyanto, M.Kom.**  
NIP 19650310 199102 1 001



## **LEMBAR PERSETUJUAN**

**RANCANG BANGUN APLIKASI KOPERASI SIMPAN  
PINJAM DENGAN METODE VIEWPOINT ORIENTED  
REQUIREMENT DEFINITION. (Studi kasus: Koperasi  
Karyawan PT Wonosari Jaya)**

### **TUGAS AKHIR**

**Disusun untuk Memenuhi Salah Satu Syarat  
Memperoleh Gelar Sarjana Komputer  
pada  
Departemen Sistem Informasi  
Fakultas Teknologi Informasi dan Komunikasi  
Institut Teknologi Sepuluh Nopember**

**Oleh:**

**Alvisha Farrasita Istifani  
0521134 0000 057**

**Disetujui Tim Penguji : Tanggal Ujian: 02 Januari 2017  
Periode Wisuda: Maret 2017**

**Sholih, S.T., M.Kom., M.SA**

  
**(Pembimbing 1)**

**Feby Artwodini Muqtadiroh, S.Kom, M.T.**

  
**(Penguji 1)**

**Tony Dwi Susanto, S.T, M.T., Ph.D.**

  
**(Penguji 2)**

# **RANCANG BANGUN APLIKASI KOPERASI SIMPAN PINJAM DENGAN METODE VIEWPOINT ORIENTED REQUIREMENT DEFINITION. (Studi kasus: Koperasi Karyawan PT Wonosari Jaya)**

**Nama Mahasiswa : Alvisha Farrasita Istfaini**  
**NRP : 0521134 0000 057**  
**Jurusan : Sistem Informasi FTIF-ITS**  
**Dosen Pembimbing 1: Sholiq, S.T., M.Kom., M.SA**

## **ABSTRAK**

*Koperasi Karyawan PT Wonosari Jaya adalah koperasi simpan pinjam terbesar di ASEAN pada awal pembukaannya. Proses bisnis pada Koperasi Karyawan PT Wonosari Jaya meliputi pencatatan simpanan dan pinjaman karyawan, pembuatan pelaporan transaksi harian, bulanan, dan tahunan. Dalam proses pengolahan data dan transaksi yang terkomputerisasi, namun masih terdapat beberapa permasalahan. Diantaranya adalah masih terdapat beberapa data yang masih salah dan belum optimal. Proses transaksi simpan pinjam dan proses pengolahan data karyawan masih menggunakan buku dan menggunakan Microsoft Excel. Karena proses yang bersifat manual tersebut, maka karyawan masih sering salah dalam input data dan juga kesulitan dalam mengumpulkan data. Dengan bertambahnya jumlah data maka akan dibutuhkan sebuah aplikasi yang dapat memudahkan pengguna dalam mengakses data. Desain dari Aplikasi Koperasi Karyawan PT Wonosari Jaya dibuat untuk memfasilitasi karyawan dalam melakukan transaksi simpan pinjam agar manajemen koperasi dapat berjalan lebih efektif dan efisien. Analisis dan Perancangan Sistem Informasi Koperasi PT Wonosari Jaya menggunakan metode Iconix Process. Tahap awal metode Iconix Process yaitu requirement analysis, peneliti*

*menggunakan metode VORD (Viewpoint Oriented Requirement Definition) untuk menganalisa kebutuhan dan desain aplikasi Koperasi Karyawan PT Wonosari Jaya. Selanjutnya dilanjutkan pada tahap Analysis & Preliminary Design, Design & Coding, dan Testing and Requirements Traceability. Hasil yang diharapkan dari tugas akhir ini berupa aplikasi koperasi, dokumen kebutuhan, desain dan perancangan perangkat lunak koperasi karyawan PT Wonosari Jaya.*

*Kata kunci: Koperasi Karyawan, Analisa Kebutuhan, Desain Aplikasi, VORD, Iconix Process*



# **RANCANG BANGUN APLIKASI KOPERASI SIMPAN PINJAM DENGAN METODE VIEWPOINT ORIENTED REQUIREMENT DEFINITION. (Studi kasus: Koperasi Karyawan PT Wonosari Jaya)**

**Nama Mahasiswa** : Alvisha Farrasita Istfaini  
**NRP** : 0521134 0000 057  
**Jurusan** : Sistem Informasi FTIF-ITS  
**Dosen Pembimbing 1:** Sholiq, S.T., M.Kom., M.SA

## **ABSTRAK**

*Cooperative employees of PT Wonosari Jaya is the largest loan in ASEAN at the beginning of its opening. Business process in Cooperative Employees of PT Wonosari Jaya include the recording of deposits and loans, and making the reporting of transactions daily, monthly, and yearly. In the process of data processing and computerized transactions, but there are still some problems. Among them are there is still some data that is still wrong and not optimal. Process transactions and loan processing and save employee data are still using books and using Microsoft Excel. Because the process is manual, then the employee is still often wrong in data input and also the difficulty in collect the data. With the increasing amount of data will be needed an app that can make it easier for users to access the data. Design of application of Cooperative Employees of PT Wonosari Jaya created to facilitate employees in conducting transactions save loan in order for cooperative management can run more effectively and efficiently. Analysis and design of information systems PT Wonosari Jaya use the Iconix Process method . The early stages of the method of Iconix Process is the requirement analysis, the researchers used a method of VORD (Viewpoint Oriented Requirement Definition) to analyze the needs and application design Cooperative Employees of PT Wonosari Jaya. Next proceed on stage Analysis Of A Preliminary Design &, Design & Coding, and Testing and Requirements Traceability. The expected results of this final*

*task form the application of the cooperative, document the needs, design and drafting software cooperative employees of PT Wonosari Jaya.*

*Keywords: cooperative employees, needs analysis, design applications, VORD, The Iconix Process*

## KATA PENGANTAR

Syukur Alhamdulillah dipanjatkan oleh peneliti atas segala petunjuk, pertolongan, kasih sayang, dan kekuatan yang diberikan oleh Allah SWT. Hanya karena ridho-Nya, peneliti dapat menyelesaikan laporan Tugas Akhir, dengan judul **RANCANG BANGUN APLIKASI KOPERASI SIMPAN PINJAM DENGAN METODE VIEWPOINT ORIENTED REQUIREMENT DEFINITION. (Studi kasus: Koperasi Karyawan PT Wonosari Jaya)**

Pada kesempatan ini, saya ingin menyampaikan banyak terima kasih kepada semua pihak yang telah memberikan dukungan, bimbingan, arahan, bantuan, dan semangat dalam menyelesaikan tugas akhir ini, yaitu kepada:

- Salam dan salawat selalu tercurah kepada junjungan kita baginda Rasulullah SAW, yang telah membawa manusia dari alam jahiliyah menuju alam yang berilmu seperti sekarang ini.
- Bapak Ir. Aris Tjahyanto, M. Kom., M. Eng. selaku Kepala Departemen Sistem Informasi.
- Bapak Sholiq, S.T ,M.Kom, M.SA, selaku dosen pembimbing yang senantiasa meluangkan waktu, memberikan saran, motivasi, dan ilmunya selama pengerjaan tugas akhir.
- Bapak Bapak Bakti Cahyo Hidayanto, S.Si., M.Kom., selaku dosen wali yang senantiasa memberikan pengarahan selama penulis menempuh masa perkuliahan dan pengerjaan tugas akhir ini.
- Bapak Hermono, selaku admin laboratorium MSI yang membantu penulis dalam hal administrasi penyelesaian tugas akhir ini.
- Kedua Orang Tua penulis, Achmed Vaival Istiadi dan Siti Cholifah yang tak hentinya memberikan doa, motivasi, dan dukungan selama pengerjaan tugas akhir.

- Saudara kandung penulis, Alvian Reynaldi Istifaiq dan Alvadyo Ghazi Istifaiq yang turut mendoakan dan mendukung penyelesaian tugas akhir.
- Tetuko Lugas Edhita Praja, yang telah setia menemani dan mendukung penulis dalam penyelesaian tugas akhir ini.
- Cahya Yunita, yang telah setia menemani dan mendukung penulis dalam penyelesaian tugas akhir ini.
- Teman – teman Lab MSI dan BELTRANIS yang telah memberikan semangat dalam menyelesaikan tugas akhir
- “Keong Club” yang beranggotakan Cahya, Sarah, Firzah, Mahesti, Niswati, RR, Selina, Yurah, Astrid, Fian dan Orie serta “Istri Idaman” yang beranggotakan Galih, Bela, Firda, Apip, dan Sheila dan juga Harun Rizal, Stezar Priansya, dan Putra yang selalu memberikan semangat dan menemani penulis saat mengerjakan tugas akhir ini.
- Serta pihak lain yang telah mendukung dan membantu dalam kelancaran penyelesaian tugas akhir ini.

Penyusunan laporan ini masih jauh dari sempurna, untuk itu peneliti menerima kritik dan saran yang membangun untuk perbaikan di masa mendatang. Penelitian ini diharapkan dapat menjadi salah satu acuan bagi penelitian – penelitian yang serupa dan bermanfaat bagi pembaca.

Surabaya, 05 Januari 2017

Penulis

## DAFTAR ISI

ABSTRAK .....	vii
ABSTRAK .....	ix
KATA PENGANTAR .....	xi
DAFTAR ISI .....	xiii
DAFTAR TABEL .....	xvii
DAFTAR GAMBAR .....	xxi
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Perumusan Masalah.....	4
1.3 Batasan Masalah.....	5
1.4 Tujuan Tugas Akhir .....	5
1.5 Manfaat Tugas Akhir .....	6
1.6 Relevansi .....	6
BAB II TINJAUAN PUSTAKA.....	9
2.1 Penelitian Sebelumnya .....	9
2.2 Dasar Teori.....	13
2.2.1 Viewpoint.....	13
2.2.1.1 Viewpoint Oriented Requirement Definition .....	14
2.2.2 UML18	
2.2.3 Iconix Process .....	19
2.2.3.1 Requirement Analysis .....	21
2.2.3.2 Analysis & Preliminary Design.....	26
2.2.3.3 Design & Coding.....	27
2.2.4 Website.....	31
2.2.5 PHP 32	
2.2.6 Code Igniter.....	33
2.2.7 Koperasi .....	34
2.2.7.1 Koperasi Simpan Pinjam .....	35
2.2.8 Profil Perusahaan PT Wonosari Jaya .....	36

BAB III METODOLOGI PENELITIAN .....	39
3.1 Requirement Analysis .....	40
3.1.1 Penggalian Kebutuhan.....	40
3.1.2 Domain Model.....	42
3.1.3 GUI Storyboard .....	43
3.1.4 Use Case Diagram .....	43
3.1.5 Use Case Description .....	43
3.2 Analysis and Preliminary Design .....	43
3.2.1 Robustness Diagram .....	43
3.3 Design & Coding.....	44
3.3.1. Sequence Diagram.....	44
3.3.2. Class Diagram .....	44
3.4 Testing .....	44
BAB IV PERANCANGAN .....	45
4.1 Requirement Analysis .....	45
4.1.1 Penggalian Kebutuhan.....	45
4.1.1.1 Wawancara .....	45
4.1.1.2 Translasi Hasil Wawancara Kedalam Bentuk Cerita Pengguna .....	48
4.1.1.3 Membuat Daftar Kebutuhan Aplikasi.....	51
4.2.2 Domain Model.....	72
4.2.3 GUI Storyboard .....	74
4.1.4 Use Case .....	82
4.1.4.1 Deskripsi Use Case.....	82
4.1.4.2 Aktor-aktor yang terlibat .....	84
4.1.4.3 Memodelkan Use Case .....	85
4.1.4.4 Use Case Diagram .....	94
4.1.4.5 Use Case Description.....	96
4.2 Analysis and Preliminary Design .....	100
4.3 Sequence Diagram.....	101
4.4 Class Diagram .....	103
BAB V IMPLEMENTASI .....	107
5.1 Implementasi .....	107
5.1.1 Pembuatan database.....	107
5.2.2 Pengkodean/Implementasi.....	108
5.2 Sampel Implementasi Aplikasi.....	110

5.3 Pengujian.....	111
BAB VI HASIL DAN PEMBAHASAN .....	155
6.1 Aplikasi Koperasi.....	155
6.2 Validasi Aplikasi.....	156
6.3 Verifikasi Aplikasi .....	156
BAB VII KESIMPULAN DAN SARAN .....	157
7.1 Kesimpulan .....	157
7.2 Saran .....	158
DAFTAR PUSTAKA .....	159
BIODATA PENULIS .....	161
LAMPIRAN A .....	A - 1 -
A.1 Requirement Analysis .....	A - 1 -
A.2 Use Case Description .....	A - 7 -
LAMPIRAN B .....	B - 1 -
B.1 Robustness Diagram Sistem Informasi Koperasi PT Wonosari Jaya .....	B - 1 -
B.2 Sequence Diagram Sistem Informasi Koperasi PT Wonosari Jaya .....	B - 60 -
LAMPIRAN C .....	C - 1 -
C.1 HASIL WAWANCARA .....	C - 1 -
C.2 DOKUMEN VALIDASI APLIKASI.....	C - 2 -
C.3 ACCEPTANCE CHECKLIST .....	C - 7 -
LAMPIRAN D .....	D - 1 -
Bukti Pendukung .....	D - 1 -



*“Halaman ini sengaja dikosongkan”*

## DAFTAR TABEL

Tabel 2.1 Penelitian 1 .....	9
Tabel 2.2 Penelitian 2 .....	11
Tabel 4.1 User Story .....	49
Tabel 4.2 Kebutuhan Fungsional .....	52
Tabel 4.3 Pendefinisian Requirement .....	54
Tabel 4.4 Deskripsi Viewpoint.....	57
Tabel 4.5 Direct Viewpoint.....	60
Tabel 4.6 Indirect Viewpoint .....	67
Tabel 4.7 Analisa Kebutuhan beserta deskripsi .....	68
Tabel 4.8 Pengelompokkan Requirement dengan Viewpoint	69
Tabel 4.9 Turunan Kebutuhan Fungsional menjadi Use Case	86
Tabel 4.10 Detail UseCase .....	88
Tabel 4.11 Use Case Description Login.....	96
Tabel 4.12 Use Case Description "Tambah Data Simpanan"	98
Tabel 5.1 Test Case Login.....	112
Tabel 5.2 Test Case Logout.....	115
Tabel 5.3 Test Case Display Data Simpanan .....	116
Tabel 5.4 Testcase Display Data Tarikan.....	117
Tabel 5.5 Testcase Display Data Pinjaman .....	118
Tabel 5.6 Testcase Display Data Angssruan .....	119
Tabel 5.7 Testcase Cari Data Simpanan.....	120
Tabel 5.8 Test Case Cari Data Pinjaman.....	121
Tabel 5.9 Test Case Cari Data Tarikan .....	122
Tabel 5.10 Test Case Cari Data Angsuran .....	123
Tabel 5.11 Edit Data Anggota.....	124
Tabel 5.12 Test Case Edit Data Simpanan .....	126
Tabel 5.13. Test Case Edit Data Tarikan.....	127
Tabel 5.14 Test Case Edit Data Pinjaman.....	129
Tabel 5.15 Test Case Verifikasi Data Simpanan.....	130
Tabel 5.16 Test Case Display Data Pelunasan Pinjaman .....	131
Tabel 5.17 Test Case Cari Data Pelunasan Pinjaman .....	132
Tabel 5.18 Test Case Unduh Data Simpanan.....	133
Tabel 5.19 Test Case Unduh Data Angsuran .....	134
Tabel 5.20 Test Case Unduh Data Pinjaman.....	135
Tabel 5.21 Test Case Unduh Data Tarikan .....	136

Tabel 5.22 Test Case Display Data Anggota.....	137
Tabel 5.23 Test Case Verifikasi Data Tarikan .....	138
Tabel 5.24 Test Case Verifikasi Data Pinjaman.....	139
Tabel 5.25 Test Case Cari Data Anggota .....	140
Tabel 5.26 Test Case Display Data Simpanan Wajib.....	142
Tabel 5.27 Test Case Cari Data Simpanan Wajib .....	143
Tabel 5.28 Test Case Update Data Simpanan Wajib.....	144
Tabel 5.29 Test Case Tambah Data Simpanan.....	145
Tabel 5.30 Test Case Tambah Data Tarikan .....	146
Tabel 5.31 Test Case Tambah Data Pinjaman.....	148
Tabel 5.32 Test Case Tambah Data Anggota .....	150
Tabel 5.33 Test Case Tambah Daftar Tagihan Angsuran.....	152
Tabel 5.34 Test Case Edit Akun .....	153
Tabel A.1 UCD Login .....	A - 7 -
Tabel A.2 UCD Logout .....	A - 8 -
Tabel A.3 UCD Display Data Simpanan.....	A - 8 -
Tabel A.4 UCD Display Data Tarikan .....	A - 10 -
Tabel A.5 UCD Display Data Pinjaman.....	A - 11 -
Tabel A.6 UCD Display Data Angsuran .....	A - 12 -
Tabel A.7 UCD Cari Data Simpanan .....	A - 13 -
Tabel A.8 UCD Cari Data Pinjaman .....	A - 14 -
Tabel A.9 UCD Cari Data Tarikan.....	A - 15 -
Tabel A.10 UCD Cari Data Angsuran.....	A - 16 -
Tabel A.11 UCD Edit Data Anggota.....	A - 17 -
Tabel A.12 UCD Edit Data Simpanan.....	A - 18 -
Tabel A.13 UCD Edit Data Tarikan .....	A - 19 -
Tabel A.14 UCD Edit Data Pinjaman .....	A - 20 -
Tabel A.15 UCD Verifikasi Data Simpanan .....	A - 21 -
Tabel A.16 UCD Display Data Pelunasan Pinjaman.....	A - 21 -
Tabel A.17 UCD Cari Data Pelunasan Pinjaman .....	A - 22 -
Tabel A.18 UCD Unduh Data Simpanan .....	A - 23 -
Tabel A.19 UCD Unduh Data Angsuran.....	A - 24 -
Tabel A.20 UCD Unduh Data Pinjaman .....	A - 25 -
Tabel A.21 UCD Unduh Data Tarikan.....	A - 26 -
Tabel A.22 UCD Display Data Anggota .....	A - 27 -
Tabel A.23 UCD Verifikasi Data Tarikan.....	A - 28 -
Tabel A.24 UCD Verifikasi Data Pinjaman .....	A - 29 -

Tabel A.25 UCD Cari Data Anggota .....	A - 30 -
Tabel A.26 UCD Display Data Simpanan Wajib.....	A - 31 -
Tabel A.27 UCD Cari Data Simpanan Wajib .....	A - 32 -
Tabel A.28 UCD Update Data Simpanan Wajib.....	A - 33 -
Tabel A.29 UCD Tambah Data Simpanan.....	A - 34 -
Tabel A.30 UCD Tambah Data Tarikan .....	A - 35 -
Tabel A.31 UCD Tambah Data Pinjaman.....	A - 36 -
Tabel A.32 UCD Tambah Data Anggota .....	A - 38 -
Tabel A.33 UCD Tambah Daftar Tagihan Angsuran....	A - 39 -
Tabel A.34 UCD Edit Akun .....	A - 40 -
Tabel B.1 RB Login .....	B - 1 -
Tabel B.2 RB Logout .....	B - 3 -
Tabel B.3 RB Display Data Simpanan.....	B - 4 -
Tabel B.4 RB Display Data Tarikan .....	B - 5 -
Tabel B.5 RB Display Data Pinjaman.....	B - 6 -
Tabel B.6 RB Display Data Angsuran .....	B - 7 -
Tabel B.7 RB Display Data Simpanan Wajib .....	B - 8 -
Tabel B.8 RB Cari Data Simpanan Wajib.....	B - 9 -
Tabel B.9 RB Cari Data Simpanan .....	B - 11 -
Tabel B.10 RB Cari Data Pinjaman .....	B - 13 -
Tabel B.11 RB Cari Data Tarikan.....	B - 15 -
Tabel B.12 RB Cari Data Angsuran.....	B - 17 -
Tabel B.13 RB Edit Data Anggota.....	B - 19 -
Tabel B.14 RB Edit Data Simpanan.....	B - 21 -
Tabel B.15 RB Edit Data Tarikan .....	B - 23 -
Tabel B.16 RB Edit Data Pinjaman .....	B - 26 -
Tabel B.17 RB Verifikasi Data Simpanan .....	B - 28 -
Tabel B.18 RB Display Data Pelunasan Pinjaman.....	B - 29 -
Tabel B.19 RB Cari Data Pelunasan Pinjaman .....	B - 31 -
Tabel B.20 RB Unduh Data Simpanan .....	B - 32 -
Tabel B.21 RB Unduh Data Angsuran.....	B - 34 -
Tabel B.22 RB Unduh Data Pinjaman .....	B - 36 -
Tabel B.23 RB Unduh Data Tarikan.....	B - 38 -
Tabel B.24 RB Display Data Anggota .....	B - 40 -
Tabel B.25 RB Verifikasi Data Tarikan.....	B - 42 -
Tabel B.26 RB Verifikasi Data Pinjaman .....	B - 43 -
Tabel B.27 RB Cari Data Anggota.....	B - 45 -

Tabel B.28 RB Update Data Simpanan Wajib .....	B - 46 -
Tabel B.29 Tambah Data Simpanan .....	B - 48 -
Tabel B.30 RB Tambah Data Tarikan .....	B - 50 -
Tabel B.31 RB Tambah Data Pinjaman .....	B - 52 -
Tabel B.32 RB Tambah Data Anggota .....	B - 54 -
Tabel B.33 RB Tambah Daftar Tagihan Angsuran .....	B - 57 -
Tabel B.34 RB Edit Akun .....	B - 58 -

## DAFTAR GAMBAR

Gambar 1.1 Roadmap Laboratorium Manajemen Sistem Informasi .....	7
Gambar 2.1 ICONIX Process.....	20
Gambar 2.2 Contoh Usecase Diagram [11].....	23
Gambar 2.3 Contoh Sequence Diagram [17] .....	29
Gambar 4.1 Proses Bisnis Transaksi Pinjaman .....	47
Gambar 4.2 Proses Bisnis Transaksi Simpanan .....	48
Gambar 4.3 Viewpoint Structuring .....	56
Gambar 4.4 Viewpoint Documentation.....	72
Gambar 4.5 Domain Model.....	73
Gambar 4.6 GUI Storyboard Simpanan .....	75
Gambar 4.7 GUI Storyboard Tarikan.....	76
Gambar 4.8 GUI Storyboard Pinjaman .....	77
Gambar 4.9 GUI Storyboard Angsuran.....	78
Gambar 4.10 GUI Storyboard Tampilan.....	79
Gambar 4.11 Halaman Login .....	80
Gambar 4.12 GUI Data Simpanan Wajib.....	81
Gambar 4.13 GUI Form Tambah Simpanan .....	81
Gambar 4.14 Form Tambah Pengguna.....	82
Gambar 4.15 Generalisasi Aktor .....	85
Gambar 4.16 Use Case Diagram Keseluruhan .....	94
Gambar 4.17 Use Case Diagram Pengguna .....	95
Gambar 4.18 Use case diagram anggota .....	95
Gambar 4.19 Use case diagram Divisi SP.....	95
Gambar 4.20 Use case diagram Ketua .....	96
Gambar 4.21 Use case diagram Bendahara.....	96
Gambar 4.22 Robustness Diagram Login .....	100
Gambar 4.23 Robustness Tambah Tagihan Angsuran .....	101
Gambar 4.24. Sequence Diagram Display Data Pinjaman...	102
Gambar 4.25 Sequence Diagram Login .....	102
Gambar 4.26 Sequence Diagram Logout .....	103
Gambar 4.27 Class Diagram (1).....	104
Gambar 4.28 Class Diagram (2).....	105
Gambar 5.1 Database Sistem Informasi Koperasi Karyawan PT Wonosari Jaya .....	107

Gambar 5.2 Pengkodean Aplikasi .....	109
Gambar 5.3 Pengkodean Aplikasi (1).....	109
Gambar 5.4 Pengkodean Aplikasi (2).....	110
Gambar 5.5 Pengkodean Aplikasi (3).....	110
Gambar 5.6 Interface Login.....	110
Gambar 5.7 Interface Tambah Simpanan .....	111
Gambar 5.8 Interface Tambah Tarikan.....	111
Gambar 6.1 Interface Lihat Data Simpanan .....	155
Gambar 6.2 Interface Transaksi Pinjaman .....	155
Gambar 6.3 Interface Form Tambah Simpanan .....	156
Gambar 6.4 Interface Form Transaksi Pinjaman.....	156
Gambar B.1 Sequence Login.....	- 60 -
Gambar B.2 Sequence Logout.....	- 61 -
Gambar B.3 Display Data Simpanan.....	- 62 -
Gambar B.4 Display Data Pinjaman.....	- 63 -
Gambar B.5 Display Data Tarikan .....	- 64 -
Gambar B.6 Display Data Angsuran .....	- 65 -
Gambar B.7 Cari Data Simpanan .....	- 67 -
Gambar B.8 Cari Data Tarikan.....	- 69 -
Gambar B.9 Cari Data Pinjaman .....	- 71 -
Gambar B.10 Cari Data Angsuran.....	- 72 -
Gambar B.11 Edit Akun.....	- 75 -
Gambar B.12 Tambah Data Anggota .....	- 77 -
Gambar B.13 Display Data Anggota.....	- 78 -
Gambar B.14 Cari Data Anggota .....	- 81 -
Gambar B.15 Display Data Pelunasan Pinjaman .....	- 82 -
Gambar B.16 Cari Data Pelunasan Pinjaman .....	- 83 -
Gambar B.17 Edit Data Simpanan .....	- 85 -
Gambar B.18 Unduh Data Simpanan .....	- 86 -
Gambar B.19 Unduh Data Angsuran.....	- 87 -
Gambar B.20 Verifikasi Data Pinjaman .....	- 88 -
Gambar B.21 Verifikasi Data Tarikan.....	- 89 -
Gambar B.22 Display Data Anggota.....	- 90 -
Gambar B.23 Cari Data Anggota .....	- 93 -
Gambar B.24 Verifikasi Data Pinjaman .....	- 94 -
Gambar B.25 Verifikasi Data Tarikan.....	- 95 -
Gambar B.26 Display Data Anggota.....	- 96 -



Gambar B.27 Cari Data Anggota .....	- 99 -
Gambar B.28 Cari Data Simpanan Wajib .....	- 100 -
Gambar B.29 Display Data Simpanan Wajib.....	- 101 -
Gambar B.30 Unduh Data Pinjaman.....	- 102 -
Gambar B.31 Unduh Data Tarikan .....	- 103 -
Gambar B.32 Tambah Data Simpanan.....	- 104 -
Gambar B.33 Tambah Data Tarikan .....	- 105 -
Gambar B.34 Tambah Data Pinjaman.....	- 107 -
Gambar C.1 Hasil Wawancara .....	- 1 -
Gambar C.2 Dokumen Validasi Aplikasi.....	- 6 -
Gambar C.3 Acceptance Checklist.....	- 7 -
Gambar D.1 Demo Aplikasi.....	- 1 -
Gambar D.2 Dokumentasi Koperasi .....	- 1 -

*“Halaman ini sengaja dikosongkan”*

# **BAB I**

## **PENDAHULUAN**

Pada bab pendahuluan ini, akan dijelaskan mengenai sekilas keadaan organisasi, masalah yang menyebabkan studi kasus ini diangkat menjadi tugas akhir, rumusan masalah dari tugas akhir ini, tujuan, dan manfaat yang dapat diambil dari *output* tugas akhir, relevansi, serta sistematika penulisan tugas akhir dengan matakuliah yang ada di Jurusan Sistem Informasi.

### **1.1 Latar Belakang**

Dewasa ini, perkembangan sistem informasi (SI) menjadi topik yang hangat diperbincangkan dan merupakan faktor yang penting bagi sebuah perusahaan untuk dapat bersaing dalam dunia perekonomian. Tidak hanya itu, penggunaan SI juga dapat meningkatkan efisiensi dan efektifitas bisnis dalam perusahaan. [1]

Begitu pula pada sektor koperasi, dimana Sistem informasi sudah bukan menjadi sarana penunjang, melainkan suatu kebutuhan yang dapat membantu pengelolaan data dan dapat membantu memperluas pasar koperasi. [2]

Bahkan tidak sedikit perusahaan yang telah menjadikan SI sebagai jantung operasional, seperti aktivitas yang terkait dengan komunikasi antara bagian internal maupun eksternal perusahaan dengan menggunakan sistem informasi. Begitu juga yang terjadi pada PT Wonosari Jaya yang merupakan Industri Kawat Baja yang pada awal pembukaannya merupakan terbesar di ASEAN. Seiring berkembangnya zaman, terbentuklah salah satu organisasi sosial pada perusahaan yaitu Koperasi Karyawan PT. Wonosari Jaya. Koperasi ini membuka beberapa jenis aktivitas untuk meningkatkan laba usahanya, yaitu berupa pinjaman regular dan juga menghimpun dana dari pihak internal maupun eksternal untuk mencapai tujuan organisasi.

Terdapat beberapa permasalahan mengenai koperasi yang dimiliki oleh PT Wonosari Jaya, yaitu perusahaan masih belum menerapkan sistem informasi pada koperasi simpan pinjam. Diantaranya adalah apabila karyawan ingin melakukan pinjaman masih harus datang ke kantor untuk mengambil form pinjaman. Selain itu, pada koperasi PT Wonosari Jaya proses pengolahan data koperasi dilakukan dengan menggunakan metode manual, maka sering terjadi kesalahan pencatatan, kesulitan dalam pencarian data, dan lain sebagainya.

Permasalahan tersebut mendorong adanya suatu kebutuhan penting untuk melakukan pengolahan data yang diharapkan dapat dikelola secara lebih efektif dan efisien. Hal ini diharapkan dapat meningkatkan kinerja koperasi khususnya pada pengolahan data transaksi simpan pinjam agar tidak terjadi kesalahan dalam pencatatan dan juga dapat mempermudah penyajian data simpan pinjam serta memudahkan dalam proses pencarian data.

Untuk itu, pada tugas akhir ini menggunakan metode (*Viewpoint Oriented Requirements Definition*) dalam pendefinisian kebutuhan. Metode ini digunakan karena didalam koperasi terdapat beberapa *viewpoint* yang memiliki kebutuhan yang berbeda-beda. Diantaranya adalah :

1. Anggota koperasi yang hanya dapat melakukan tambah transaksi, melihat dan mencari data transaksi.
2. Divisi SP memiliki peranan penting dalam koperasi karyawan PT Wonosari Jaya, diantaranya adalah membuat akun pengguna, melakukan verifikasi transaksi, melakukan unduh data, melakukan display dan cari data transaksi serta edit data transaksi
3. Ketua Koperasi memiliki tugas dalam verifikasi data transaksi, serta melakukan display dan cari data transaksi serta edit data transaksi

4. Bendahara, dapat melakukan verifikasi data transaksi, update data simpanan wajib, unduh data transaksi serta display dan cari data transaksi serta edit data transaksi

Karena itu, penggunaan metode VORD dalam penggalian kebutuhan dapat membantu peneliti dalam menggambarkan kebutuhan sistem sebagai suatu layanan yang dapat diberikan oleh sistem kepada viewpoint (sudut pandang) sebagai pengguna sistem dan juga berguna dalam mendefinisikan proses bisnis perusahaan yang didalamnya terdapat beberapa aktor yang memiliki kebutuhan masing-masing. Sehingga, penggunaan metode VORD ini diharapkan dapat memetakan secara detail setiap aktor yang berhubungan dengan kebutuhan masing-masing. VORD merupakan salah satu metode yang digunakan untuk mendefinisikan kebutuhan dengan pendekatan *viewpoint* (sudut pandang) dari pengguna sistem. [3] Kelebihan dari metode VORD ini sendiri adalah VORD dapat mengatasi kebutuhan dari berbagai perspektif pengguna sehingga menciptakan kerangka kerja untuk membedakan kebutuhan pengguna. [4]

Namun, masih diperlukan suatu disiplin dan pemodelan yang baik agar menghasilkan aplikasi yang memenuhi kebutuhan koperasi. Maka, peneliti menggunakan *Unified Modelling Language* (UML) sebagai notasi untuk menggambarkan dan mendokumentasikan sistem. Salah satu metodologi yang sesuai bisnis proses koperasi karyawan yang cenderung berubah-ubah ini adalah ICONIX Process. Pemilihan metode ICONIX Process sangat membantu peneliti karena metode ICONIX Process bersifat iterative, bertahap dan didalamnya terdapat robustness diagram yang dapat menjembatani *gap* antara desain dan pengkodean.

ICONIX Process berada diantara dua pendekatan rekayasa perangkat lunak yaitu Rational Unified Process (RUP) dengan proses perancangan yang sangat banyak dan Extreme Programming (XP) dengan proses perancangan yang sangat sedikit. Dalam

perancangannya Use case memegang peran penting dalam mengendalikan ICONIX Process seperti halnya RUP, namun ICONIX Process hanya memiliki proses yang singkat seperti halnya XP. Dalam rekayasa perangkat lunak menggunakan ICONIX Process yang singkat ini tidak mengabaikan desain dan analisis, penggunaan UML pada ICONIX process lebih efisien karena berfokus terhadap requirement [5].

Berdasarkan permasalahan di atas, maka tujuan penelitian ini adalah untuk menghasilkan suatu sistem yang dapat meningkatkan efektifitas dan efisiensi dalam pengolahan data transaksi simpan pinjam. Fokus utama pada penelitian ini adalah perancangan kebutuhan dengan menggunakan metode VORD serta pengembangan dan pendokumentasian sistem dengan menggunakan ICONIX Process.

Manfaat yang dapat diperoleh dari penelitian ini adalah dapat membantu anggota serta pegawai koperasi dalam melakukan pengolahan data dan membantu anggota PT Wonosari Jaya dalam melakukan simpan pinjam dengan menggunakan sistem informasi.

## **1.2 Perumusan Masalah**

Berdasarkan latar belakang yang telah dijabarkan di atas rumusan masalah yang dijadikan acuan dalam pengerjaan tugas ini adalah:

1. Seperti apa hasil analisa spesifikasi kebutuhan perangkat lunak berdasarkan metode VORD (Viewpoint Oriented Requirement Definition dapat membantu dalam mengevaluasi program kerja pada Koperasi Karyawan PT Wonosari Jaya?
2. Seperti apa hasil desain perangkat lunak berdasarkan metode ICONIX Process dapat membantu dalam mengevaluasi program kerja pada Koperasi Karyawan PT Wonosari Jaya?

3. Seperti apa hasil akhir perangkat lunak Koperasi Karyawan PT Wonosari Jaya?

### **1.3 Batasan Masalah**

Dalam pengerjaan tugas akhir ini, ada beberapa batasan masalah yang harus diperhatikan, yaitu sebagai berikut:

1. Studi kasus yang digunakan hanya pada ruang lingkup koperasi PT Wonosari Jaya
2. Sistem hanya membahas bidang usaha simpan pinjam
3. Aplikasi program berisi data anggota, simpanan, pinjaman, tarikan, dan angsuran.
4. Pada tahap analisa kebutuhan, hanya menganalisa kebutuhan fungsional saja hingga pada tahap pengujian.
5. Studi kasus berada di Jawa Timur.
6. Tahapan pengerjaan tugas akhir ini dikerjakan mulai tahap analisa kebutuhan hingga tahap testing perangkat lunak
7. Tahap pengerjaan tugas akhir ini tidak termasuk dalam tahap pemberian pelatihan penggunaan perangkat lunak kepada pengguna akhir.

### **1.4 Tujuan Tugas Akhir**

Berdasarkan hasil perumusan masalah dan batasan masalah yang telah disebutkan sebelumnya, maka tujuan yang dicapai dari tugas akhir ini adalah sebagai berikut:

1. Untuk mendapatkan spesifikasi kebutuhan dan mendesain suatu sistem informasi pada koperasi karyawan PT Wonosari Jaya
2. Untuk memberikan kemudahan pada pengguna sistem informasi pada koperasi karyawan PT Wonosari Jaya



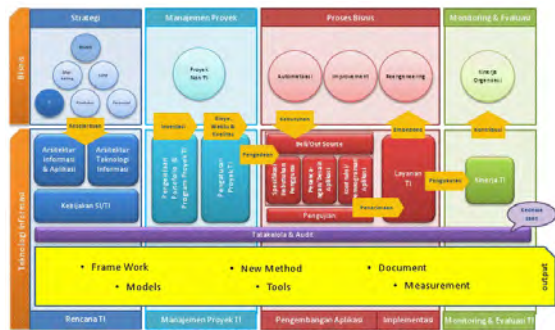
## **1.5 Manfaat Tugas Akhir**

Manfaat yang dapat diperoleh dari pengerjaan tugas akhir ini adalah sebagai berikut:

1. Membantu perusahaan untuk meningkatkan kualitas proses analisa kebutuhan perangkat lunak dengan mendefinisikan spesifikasi kebutuhan secara jelas dengan menggunakan metode VORD.
2. Sebagai acuan dalam merancang kebutuhan secara detail untuk membuat aplikasi sistem informasi koperasi karyawan.
3. Rancangan kebutuhan ini selanjutnya dapat memberikan kemudahan pada pegawai koperasi dalam melakukan pengolahan data dan kemudahan pada karyawan PT Wonosari Jaya dalam melakukan kegiatan simpan pinjam.

## **1.6 Relevansi**

Relevansi pengerjaan tugas akhir ini berada dalam ruang lingkup penelitian pada Laboratorium Manajemen Sistem Informasi yang dimana penelitian ini termasuk pada topik Pengembangan Perangkat lunak yang selanjutnya menghasilkan sistem aplikasi perangkat lunak Koperasi Simpan Pinjam PT Wonosari Jaya. Penelitian ini juga memiliki relevansi terhadap mata kuliah wajib Analisa dan Desain Perangkat Lunak serta Konstruksi dan Pengujian Perangkat Lunak. Dibawah ini telah dijelaskan secara detail mengenai roadmap pada Laboratorium Manajemen Sistem Informasi yang menunjukkan bahwa tugas akhir ini termasuk pada Pengembangan Aplikasi yaitu pada Perancangan atau Desain Aplikasi.



**Gambar 1.1 Roadmap Laboratorium Manajemen Sistem Informasi**

*“Halaman ini sengaja dikosongkan”*

## **BAB II**

### **TINJAUAN PUSTAKA**

Bab ini akan menjelaskan mengenai penelitian sebelumnya dan dasar teori yang dijadikan acuan atau landasan dalam pengerjaan tugas akhir ini. Landasan teori akan memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini.

#### **2.1 Penelitian Sebelumnya**

Dalam penelitian ini, digunakan beberapa penelitian terdahulu sebagai pedoman dan referensi dalam melaksanakan proses-proses dalam penelitian, seperti yang terdapat pada Tabel 2.1 dan 2.2 Informasi yang disampaikan dalam tabel berikut berisi informasi penelitian sebelumnya, hasil penelitian, dan hubungan penelitian terhadap penelitian dalam rangka tugas akhir ini.

**Tabel 2.1 Penelitian 1**

<b>Penelitian 1</b>	
Judul Penelitian	<i>Perancangan Aplikasi Pencatatan Kehadiran Kuliah Pada Smartphone Android Dengan Metode Pengembangan Iterative Parallel Prototyping Dan Iconix Process</i>
Penulis	Ashr Hafiizh Tantri
Tahun Penelitian	2017
Hasil Penelitian	Mockup user interface
Objek Penelitian	Jurusan Sistem Informasi ITS

Penelitian 1	
Metode	<ul style="list-style-type: none"> <li>- Studi Literatur <i>Input</i> dalam tahapan ini adalah paper, jurnal ilmiah, buku, dan media lainnya. Sementara <i>output</i> dari tahapan ini adalah paper, jurnal ilmiah, buku, dan media lainnya.</li> <li>- Pengumpulan Kebutuhan Awal Menganalisis kebutuhan pengguna untuk mengetahui proses bisnis pada Jurusan Sistem Informasi ITS. Sementara <i>output</i> dari tahapan ini adalah alur pencatatan kehadiran yang telah diperbarui dan tabel daftar kebutuhan awal.</li> <li>- Pembuatan Storyboard Membuat perkiraan awal dengan menggunakan kertas dan alat tulis yang dibutuhkan. Sementara <i>output</i> dari tahapan ini adalah <i>Storyboard</i></li> <li>- Pembuatan Desain UI Melakukan pembuatan purwarupa tampilan antarmuka aplikasi. Sementara <i>output</i> dari tahapan ini adalah purwarupa antarmuka pengguna</li> <li>- Pengujian Desain UI Melakukan pengujian dari desain UI yang telah dibuat sebelumnya. Sementara <i>output</i> dari tahapan ini adalah <i>usability study</i> dan <i>heuristic evaluation</i></li> <li>- Pembuatan Desain Sistem Membuat beberapa diagram yang digunakan untuk ‘menerjemahkan’ purwarupa tampilan antar muka sehingga bisa lebih mudah untuk</li> </ul>

Penelitian 1	
	<p>dijadikan kode program. Sementara <i>output</i> dari tahapan ini adalah <i>Domain Model</i>, <i>Use Case Model</i>, <i>Robustness Diagram</i>, <i>Sequence Diagram</i>, dan <i>Class Diagram</i>.</p> <ul style="list-style-type: none"> <li>- Validasi Sistem Melakukan validasi sistem dengan menggunakan <i>traceability matrix</i>. Sementara <i>output</i> dari tahapan ini adalah <i>requirement traceability matrix</i>.</li> <li>- Penyusunan Laporan TA Menyusun laporan TA dengan menggunakan hasil dari tahapan-tahapan sebelumnya. Sementara <i>output</i> dari tahapan ini adalah buku tugas akhir yang telah disusun sesuai dengan standard.</li> </ul>
Keterkaitan dengan penelitian	Hasil penelitian berupa aplikasi pencatatan kehadiran mahasiswa berbasis android
Kelebihan	Produk dokumen jelas yaitu berupa desain sistem yang sesuai dengan kebutuhan pengguna serta tahapan dalam metode penelitian dijabarkan secara jelas.
Kekurangan	Dalam pengujian <i>Heuristic Evaluation</i> belum menggunakan pakar UI sehingga hasil yang didapat belum sepenuhnya valid.

Tabel 2.2 Penelitian 2

Penelitian 2	
Judul Penelitian	<i>Analisis dan Perancangan Sistem Informasi Penerimaan Negara Bukan Pajak (PNBP) Terintegrasi pada Balai Besar Pelaksanaan Jalan Nasional V</i>

Penelitian 2	
Penulis	Endones Putra Yusa
Tahun Penelitian	2017
Hasil Penelitian	Software Sistem Informasi PNBPTerintegrasi
Objek Penelitian	Balai Besar Pelaksanaan Jalan Nasional V
Metode	<ul style="list-style-type: none"> <li>- Wawancara Melakukan wawancara kepada Bendahara dan Staff Bendahara untuk mendapatkan informasi kebutuhan fungsional yang baru</li> <li>- Studi Kepustakaan Metode yang dilakukan untuk pengumpulan data dengan cara mencari sumber-sumber literature yang terkait dengan bahasan penelitian, dengan hasil akhir yang digunakan sebagai landasan teori dalam penulisan laporan</li> <li>- Observasi Metode pengumpulan data yang dilakukan dengan cara pengamatan langsung dengan obyek yang diteliti yaitu <i>Simponi</i> dan <i>Silabi</i>.</li> <li>- Metode Analisis dan Perancangan Menggunakan metode ICONIX Process yang didalamnya terdapat empat tahapan. Pada penelitian ini hanya menggunakan 2 tahapan <i>Requirements</i>, dan <i>Detailed Design</i></li> </ul>
Keterkaitan dengan penelitian	Hasil penelitian berupa Dokumen Analisis dan Desain Perangkat Lunak dan menggunakan metode ICONIX Process

Penelitian 2	
Kelebihan	Metode penelitian yang dibuat oleh peneliti jelas serta penjabaran kebutuhan sistem dijabarkan dengan jelas.
Kekurangan	Output dari penelitian belum dijabarkan dan dijelaskan dengan jelas

## 2.2 Dasar Teori

Pada bagian ini, akan dijelaskan mengenai teori-teori yang digunakan untuk mendukung pengerjaan tugas akhir. Teori tersebut yaitu mengenai, perancangan sistem informasi, UML, Iconix Process, VORD, serta pemahaman mengenai koperasi simpan pinjam.

### 2.2.1 Viewpoint

Dalam penggalan kebutuhan, terdapat pendekatan yang berbasis pada *viewpoint* (sudut pandang) yang dimana semua informasi yang berkaitan dengan kebutuhan sistem tidak dapat ditentukan hanya dengan satu perspektif saja. Sehingga, dalam penggalan kebutuhan dibutuhkan beberapa sudut pandang. Informasi yang berasal dari berbagai macam sudut pandang selanjutnya harus diintegrasikan untuk dapat membentuk spesifikasi sistem yang sesungguhnya.

Pada pendekatan ini, terdapat dua tipe *viewpoint* yang diusulkan, diantaranya adalah : [6]

1. ***Viewpoints associated with system stakeholders***, stakeholder sebuah sistem yaitu seseorang yang secara langsung atau tidak langsung berhubungan dengan sistem. Stakeholder disini dapat berupa pengguna akhir pada sistem atau *customer* yang telah menginstall sistem.
2. ***Viewpoints associated with organisational and domain knowledge***, sebuah organisasi dan pengetahuan domain



adalah pengetahuan yang membatasi kebutuhan sistem. Contohnya adalah kinerja jaringan atau hardware yang digunakan pada divisi yang berbeda pada suatu organisasi. Tipe *viewpoint* ini tidak dapat digabungkan dengan berbagai stakeholder, tetapi model ini dapat berisi informasi yang berasal dari berbagai sumber.

Berbagai perbedaan model pada sebuah viewpoint dapat menghasilkan beberapa perbedaan perspektif dalam penggalan kebutuhan. Oleh karena itu, terdapat berbagai macam model dalam penggalan informasi. Diantaranya adalah : [6]

1. ***Viewpoints for Requirements Elicitation***, berisi proses dalam menggali kebutuhan yang didalam kebutuhan tersebut harus berisi fakta. Tahapan pada metode ini terdiri dari menggali kebutuhan, analisa kebutuhan dan merundingkan kebutuhan dengan beberapa stakeholder dari kebutuhan yang telah didapatkan.
2. ***Viewpoints for Requirements Modelling***, berisi proses yang merepresentasikan kebutuhan yang terdapat pada organisasi. Selain itu juga memodelkan kebutuhan sesuai dengan spesifikasi kebutuhan. Metode ini lebih berfokus pada pencapaian tujuan organisasi.

### **2.2.1.1 Viewpoint Oriented Requirement Definition**

VORD (Viewpoint-Oriented Requirements) Definition) adalah salah satu metode untuk menganalisis kebutuhan sistem dengan menggunakan pendekatan Viewpoint. Metode ini ini dikembangkan oleh Gerald Kotonya and Ian Sommerville pada tahun 1996. Metode ini termasuk pada model *Elicitation* yang dimana didalamnya terdiri proses *discovery*, *analysis* dan *negotiation*. Metode ini dikembangkan untuk membantu proses spesifikasi dari interaksi sistem dan juga untuk menganalisa kebutuhan perangkat lunak berdasarkan sudut pandang pengguna. [6]

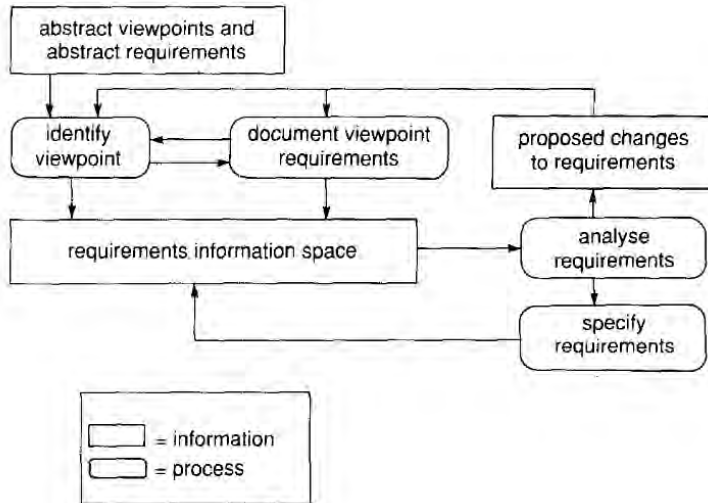
Menurut Gerald Kotonya and Ian Sommerville, hal yang terpenting dalam membuat perangkat lunak adalah pada proses analisa kebutuhan, bukan pada *coding* atau *desain* dari perangkat lunak. Sedangkan menurut Zelika, VORD sangat berguna dalam menemukan kebutuhan user dan juga berguna dalam mengidentifikasi sistem. VORD terfokus pada entitas eksternal yang berinteraksi dengan sistem. Oleh karenanya VORD merepresentasikan kebutuhan dari sistem berdasarkan entitas viewpoint. [7]

Kelebihan dari metode ini yaitu VORD menyediakan metode yang terstruktur untuk mengambil data, dokumentasi data dan analisa data untuk menspesifikasikan sudut pandang berdasarkan kebutuhan pengguna. [7]

Viewpoint terbagi menjadi direct viewpoint dan indirect viewpoint. Direct Viewpoint menggambarkan tentang entitas yang berkorespondensi secara langsung dengan pelanggan. Sedangkan Indirect Viewpoint menggambarkan entitas yang berkepentingan untuk menerima service dari sistem namun tidak berinteraksi langsung dengan sistem. [4]

1. **Direct Viewpoint** : Klien atau pengguna menerima atau menggunakan perangkat lunak secara langsung yang dimana pengguna yang bertugas untuk mengirimkan informasi dan data ke sistem atau perangkat lunak yang dibuat.
2. **Indirect Viewpoint** : Klien atau pengguna tidak memiliki interaksi dengan sistem atau tidak secara langsung menjalankan sistem. Setiap sudut pandang (*viewpoint*) memiliki hubungan dengan sistem yang berdasar pada kebutuhan dan interaksi sistem. Tetapi, jika semua *viewpoint* telah dianalisis dan dispesifikasi maka semua kebutuhan sistem juga telah dianalisis dan dispesifikasikan.

VORD memiliki empat tahap utama di dalam melakukan identifikasi kebutuhan yaitu: [8]



**Gambar 2.5 Alur Proses VORD [8]**

### 1. *Viewpoint Identification*

Identifikasi viewpoint meliputi penemuan viewpoint berdasarkan layanan yang akan diterima oleh setiap viewpoint. Di dalam tahap ini, dilakukan identifikasi terhadap kebutuhan dengan langkah sebagai berikut :

- Mendaftar semua stakeholder yang ada di dalam sistem perangkat lunak tersebut dengan melakukan brainstorming.
- Mengidentifikasi mana yang termasuk ke dalam viewpoint dan service (layanan) di dalam kebutuhan sistem perangkat lunak.

### 2. *Viewpoint Documentation*

Dokumentasi viewpoint meliputi cara untuk mendiskripsikan setiap viewpoint dan layanan yang telah ditentukan sebelumnya. Tujuan dari viewpoint documentation ini adalah untuk memetakan kebutuhan sistem atau layanan terhadap masing-masing viewpoint sesuai dengan viewpoint structuring. Pemetaan

kebutuhan tersebut meliputi kebutuhan fungsional dan non-fungsional sistem. Hasil dari viewpoint documentation digunakan sebagai acuan pembuatan use case.

### 3. *Viewpoint Analyze & Spesification*

Tahap ini merupakan tahap akhir yang meliputi analisa kebutuhan pada setiap viewpoint yang telah didokumentasikan, setelah itu dilakukan spesifikasi dengan menggunakan *matrix* untuk menghubungkan setiap viewpoint dengan kebutuhan yang telah didokumentasikan sebelumnya.

VORD merupakan metode analisa kebutuhan yang berdasar pada pemahaman kebutuhan yang tidak dilihat sebagai sesuatu yang sama bagi *stakeholder* yang berbeda-beda. Konsumen yang membayar untuk sistem memiliki berbagai macam perspektif pada kebutuhan yang sistem yang berbeda-beda. Sistem yang kompleks memiliki banyak sub-komponen, yang dimana user yang berbeda-beda akan mengartikulasikan persyaratan dengan beragam spesifikasi. Sudut pandang yang berbeda sering mengakibatkan perspektif yang jelas berbeda dari masalah yang sama dan hal ini digunakan untuk membantu mengkategorikan dan membuat struktur persyaratan yang dibutuhkan oleh sistem. [7]

Selain metode VORD, terdapat metode MAMIE dan metode CREW yang dimana kedua metode ini termasuk pada model *Viewpoint Modelling* yang dimana didalamnya terdapat proses spesifikasi dan memodelkan kebutuhan yang berfokus pada tujuan organisasi. Metode CREW digunakan untuk menemukan kebutuhan sistem dengan menghubungkan tujuan (*goal*) organisasi dengan skenario (use case) yang menggambarkan perilaku sistem. [6] Disamping itu, terdapat metode MAMIE yang didalamnya juga menghubungkan antara tujuan organisasi dengan skenario yang dijabarkan dalam bentuk usecase

diagram. Metode ini digunakan untuk membantu sistem analisis dalam menggali kebutuhan sistem.

### 2.2.2 UML

Saat ini dunia industri membutuhkan sebuah framework untuk mengukur dan menguji teknik perancangan. UML (*Unified Modeling Language*) adalah metode pemodelan secara visual sebagai sarana untuk merancang dan atau membuat software berorientasi objek. Karena UML ini merupakan bahasa visual untuk pemodelan bahasa berorientasi objek, maka semua elemen dan diagram berbasiskan pada paradigma object oriented. UML sendiri juga memberikan standar penulisan sebuah sistem blueprint, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. Macam-macam diagram UML terdiri dari use case diagram, narrative usecase, activity diagram, sequence diagram. [9]

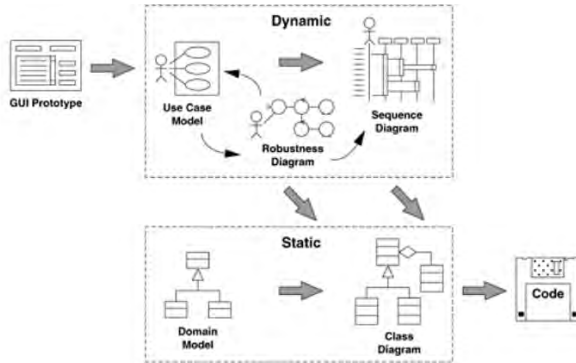
Karena tergolong bahasa visual, UML lebih mengedepankan penggunaan diagram untuk menggambarkan aspek dari sistem yang sedang dimodelkan. Memahami UML itu sebagai bahasa visual itu penting, karena penekanan tersebut membedakan dengan bahasa pemrograman yang lebih dekat ke mesin. Bahasa visual lebih dekat ke mental model pikiran kita, sehingga pemodelan menggunakan bahasa visual bisa lebih mudah dan lebih cepat dipahami dibandingkan apabila dituliskan dalam sebuah bahasa pemrograman. Seperti yang telah dipaparkan diatas, UML yang merupakan turunan dari beberapa metode mempunyai kumpulan diagram grafis sebagai kombinasi dari konsep pemodelan data (entity relationship diagram), pemodelan bisnis (work flow), pemodelan obyek, dan pemodelan komponen. Diagram grafis tersebut merupakan tampilan dari beberapa level abstraksi yang dapat digunakan secara bersama oleh semua proses pada seluruh lifecycle

pengembangan software serta pada implementasi ke beberapa teknologi yang berbeda. [9]

### 2.2.3 Iconix Process

Metode Iconix process diperkenalkan oleh Doug Rosenberg. Iconix process merupakan salah satu model rekayasa perangkat lunak yang dapat digunakan dalam pengembangan sebuah software [10] . Pendekatan Iconix Process berada diantara *Rational Unified Process* (RUP) yang sangat luas dan *eXtreme Programming* (XP) yang sangat sederhana. Iconix process tetap menggunakan use case tanpa banyak menggunakan pentabelan dan tidak mengabaikan analisis dan desain seperti XP. Hal tersebut yang menjadikan iconix process menjadi efisien karena tetap fokus pada pendefinisian kebutuhan. Maka dapat disimpulkan bahwa iconix process adalah metode pengembangan software dengan pendekatan minimalis dan efisien dari tahap use case sampai pemrograman atau *coding*.

Penelitian ini menggunakan langkah-langkah yang terdapat dalam Iconix Process sehingga sistem yang selanjutnya di desain akan tergambarkan secara jelas bahkan dalam implementasinya akan lebih mudah. Kesimpulan dalam penjelasan diatas adalah Iconix Process merupakan metode yang didalamnya berisi proses, langkah, teknik mengenai bagaimana cara agar menghasilkan kode dari suatu program. [11]



**Gambar 2.1 ICONIX Process**

Dapat dilihat dari gambar 2.1 bahwa Iconix Process memiliki dua struktur model yaitu struktur dinamis dan struktur statis. Di dalam struktur dinamis terdapat use case model, robustness diagram dan sequence diagram. Sedangkan pada struktur statis terdapat domain model dan class diagram.

Dibawah ini merupakan beberapa fitur utama pada Iconix Process : [11]

- a. Iconix Process merupakan sebuah metode yang menggunakan *use case* atau biasa disebut dengan *use case driven*. Use case tersebut harus ditentukan terlebih dahulu dari awal pengembangan sistem karena selanjutnya akan menjadi dasar dalam menentukan model dan perilaku dari sistem yang akan dibangun.
- b. Iconix process merupakan metode yang bersifat *iterative* (berulang) dan *incremental* (bertahap). Banyak iterasi yang dilalui ketika menentukan domain model, mengidentifikasi dan menganalisa use case, dan iterasi-iterasi lain yang terjadi seiring berjalannya siklus hidup pengembangan sistem. Komponen-komponen yang terlibat dalam arsitektur statis yang dihasilkan, selanjutnya diperbaiki (diupdate) secara

bertahap dengan bantuan dari komponen-komponen yang terdapat pada arsitektur dinamis yang terdiri dari *use case*, *robustness diagram*, dan *sequence diagram*.

- c. Iconix Process merupakan metode yang berdasarkan arsitektur (architecture-centric). Arsitektur yang terdapat pada Iconix Process terbagi menjadi dua yaitu arsitektur statis dan arsitektur dinamis. Dimana arsitektur statis nantinya akan menjadi kode-kode program, sedangkan arsitektur dinamis nantinya akan menggambarkan perilaku dari sistem.
- d. Iconix Process menawarkan penggunaan UML (*Unified Modelling Language*) secara tidak berlebihan karena hanya melakukan langkah-langkah yang dianggap perlu dan cukup untuk melakukan analisa dan perancangan berorientasi obyek (*Object Oriented*)
- e. Iconix Process memiliki kemampuan penelusuran (*traceability*) yang cukup tinggi. Merujuk kembali kepada kebutuhan fungsional dapat dilakukan dengan berbagai cara mudah pada setiap fase pengembangan software. Penelusuran ini juga tampak pada kenyataan setiap obyek dapat dilacak langkah demi langkah, dari analisa menjadi desain dan berakhir dengan implementasi.

### 2.2.3.1 Requirement Analysis

Dalam requirement terhadap tiga tahap yaitu : [11]

- *Functional requirements*  
Mengumpulkan segala kebutuhan fungsional yang diperlukan dalam pembuatan perangkat lunak. Kebutuhan fungsional dari perangkat lunak merupakan modal utama dalam pengembangan perangkat lunak. Semua kebutuhan dalam pengembangan perangkat lunak dikumpulkan menjadi satu bagian. Kemudian dilakukan analisis



mengenai kebutuhan fungsional dan kebutuhan non fungsional.

- *Domain modeling*

Domain modeling merupakan pondasi dari bagian static pada UML. Domain modeling didapatkan dari mengekstrak kata benda yang didapatkan dari functional requirements. Kata benda yang didapatkan saling dihubungkan sesuai kebutuhan dari perangkat lunak.

- *Behavioral Requirements Definition*

Pendefinisian behavioral requirements dilakukan dengan dua cara yaitu :

**a. Membuat GUI Storyboard**

GUI Storyboard disesuaikan dengan functional requirements yang telah didefinisikan sebelumnya. Functional requirement tersebut selanjutnya digunakan untuk merancang GUI storyboard.

**b. Use Case modeling**

Use Case modeling merupakan bagian dari Iconix Process yang menjelaskan tentang segala hal yang dilakukan oleh pengguna dari sistem. Proses ini menjelaskan tentang segala hal yang dilakukan oleh pengguna dan hubungan terhadap tanggapan dari sistem. Dalam proses ini, desain perangkat lunak diharapkan dijelaskan secara rinci karena perangkat lunak didedikasikan berdasarkan kebutuhan pengguna.

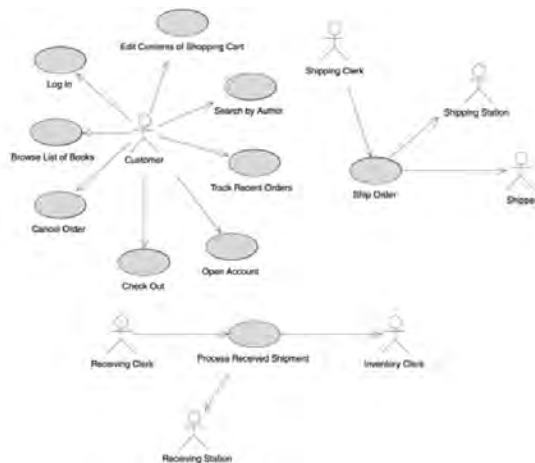
➤ **Use Case Diagram**

Use case diagram berisi mengenai interaksi antara sekelompok proses dengan sekelompok aktor, menggambarkan fungsionalitas dari sebuah sistem yang dibangun dan bagaimana sistem berinteraksi dengan dunia luar. Use case diagram dapat digunakan selama proses analisis untuk

memahami secara jelas kebutuhan sistem dan untuk memahami bagaimana sistem seharusnya bekerja. [12]

Use case sendiri merupakan sebuah *methodology* yang digunakan untuk melakukan identifikasi, mengklarifikasi, dan mengorganisasi sebuah kebutuhan system [13]. Antara *Use Case* yang satu dengan stu dengan *Use Case* lainnya memiliki hubungan saling ketrgantungan (*dependency arrow*). Usecase diagram terdiri dari 6 elemen utama, yaitu sistem, aktor, use case, asosiasi, *Dependency*, dan *Generalization*. Berikut ini adalah bentuk dari *Use Case* diagram yang ditunjukkan pada Gambar 2.2 .

Use case diagram dapat digunakan selama proses analisis untuk menangkap kebutuhan sistem dan untuk memahami bagaimana sistem seharusnya bekerja [14]. Pada diagram use case, orang – orang atau sistem yang berinteraksi dengan sistem target dinamakan actors, sedangkan fitur yang digunakan oleh actor disebut dengan Use Case. Antara Use Case satu dengan Use Case lainnya dapat memiliki hubungan yang dihubungkan dengan panah ketergantungan (*dependency arrow*).



**Gambar 2.2 Contoh Usecase Diagram [11]**

- **System**, sebuah diagram yang membatasi hubungan antara Actor yang menggunakan sistem itu (di luar sistem) dengan fitur yang disediakan sistem (di dalam sistem).
- **Actor**, menggambarkan pengguna software aplikasi (user) yang bisa berupa orang, sistem lain, maupun perangkat lainnya yang memiliki peran dalam berjalannya sistem.
- **Use Case**, menggambarkan fitur utama dari sebuah sistem, tanpa fitur ini, sistem akan gagal/tidak mampu memenuhi kebutuhan aktor (user). Setiap Use Case menunjukkan tujuan yang harus dicapai sistem.
- **Association**, menunjukkan hubungan antara aktor dengan Use Case. Setiap Association akan menjadi dialog yang harus dijelaskan dalam naratif Use Case. Setiap naratif yang akan menjadi sebuah set skenario yang berfungsi menjadi semacam test case ketika mengevaluasi analisa, desain, dan implementasi dari Use Case.
- **Dependency**, menggambarkan hubungan komunikasi antara dua Use Case
- Use Case Description

*Use case description* merupakan deskripsi tertulis mengenai bagaimana sebuah actor melakukan suatu tugas dalam sebuah aplikasi. Use case merepresentasikan tahapan yang simple, dimulai dengan user's goal dan diakhiri ketika tujuan tercapai. [15] Use case sendiri dibagi menjadi 2. Pertama, yaitu *primary course*. Pada bagian ini menjelaskan bagaimana respon system terhadap pengguna dan asumsi system berjalan lancar. Kedua, yaitu *alternate course*, dalam *alternate course* ini, menjelaskan scenario apabila scenario utama/ main scenario belum tidak dilakukan/gagal.

Use case didokumentasi dalam use case model sebagai berikut:  
[12]

1. **Use Case Name.** Nama yang diberikan pada masing-masing use case

2. **Summary.** Deskripsi singkat use case, biasanya terdiri dari satu atau dua kalimat.
3. **Dependency.** Bagian ini menggambarkan apakah use case yang satu tergantung pada use case yang lain, dalam arti apakah use case tersebut termasuk pada use case yang lain atau malah memperluas use case lain.
4. **Actors.** Bagian ini memberikan nama pada aktor dalam use case. Selalu terdapat use case utama (primary use case) yang memulai use case. Disamping itu terdapat juga secondary use case yang terlibat dalam use case. Contohnya, dalam use case Withdraw Funds, ATM, dimana Customer adalah aktor-nya.
5. **Preconditions.** Satu atau lebih kondisi harus berjalan dengan baik pada permulaan use case; contohnya mesin ATM yang tidak jalan, menampilkan pesan Selamat Datang.
6. **Deskripsi.** Bagian terbesar dari use case merupakan deskripsi naratif dari urutan utama use case yang merupakan urutan yang paling umum dari interaksi antara aktor dan sistem. Deskripsi tersebut berkaitan dengan apa yang sistem lakukan dalam merespon input aktor, bukan bagaimana internal melakukannya.
7. **Alternatif-alternatif.** Deskripsi naratif dari alternative merupakan cabang dari urutan utama. Terdapat beberapa cabang alternatif dari urutan utama. Contohnya, jika rekening customer terdapat dana yang tidak sesuai, akan tampil permohonan maaf dan menolak kartu.
8. **Postcondition.** Kondisi yang selalu terjadi di akhir usecase, jika urutan utama telah dilakukan; contohnya dana customer telah ditarik.
9. **Outstanding questions.** Pertanyaan-pertanyaan tentang use case didokumentasikan untuk didiskusikan dengan para user.

### c. Milestone 1 : Requirements Review

Dalam peninjauan ulang, hal yang dilakukan sebagai berikut :

- Memastikan bahwa bahwa use case text telah sesuai dengan kebutuhan pengguna.
- Memastikan bahwa domain model telah menunjukkan hubungan yang benar.
- Memastikan bahwa use case telah terorganisir dalam satu paket.

#### 2.2.3.2 Analysis & Preliminary Design

Pada tahap *Analysis & Preliminary Design*, terdapat tiga tahapan yang harus dilakukan diantaranya adalah : [11]

- *Robustness analysis*

Robustness Analysis digunakan untuk membantu dalam menjembatani gap (kesenjangan) dari analisis ke desain. Singkatnya, robustness merupakan cara yang digunakan untuk menganalisis teks use case dan mengidentifikasi anggapan awal dari objek untuk setiap use case. Robustness Analysis membantu memperbaiki teks use case dan domain model. Analisis ini dilakukan dengan cara membuat Robustness Diagram yang menghubungkan antara analisis dan desain. Robustness Diagram dapat memastikan bahwa use case ditulis dalam konteks domain model. Yaitu semua syarat (kata benda dan frase kata benda) yang masuk ke model domain juga harus digunakan secara langsung dalam use case description. Analisis dilakukan dengan membuat robustness diagram yang menghubungkan antara analisis dan desain.

Robustness diagram secara visual merepresentasikan perilaku use case, serta menunjukkan kelas yang berpartisipasi dan perilaku perangkat lunak.

- *Milestone 2 : Preliminary Design Review (PDR)*

Pada tahap ini dilakukan analisis mengenai kecocokan use case dengan robustness diagram dan memastikan semua entitas yang terdapat pada robustness diagram telah diperbarui pada domain model. [11]

### 2.2.3.3 Design & Coding

Pada tahap *Design & Coding*, terdapat tiga tahapan yang harus dilakukan diantaranya adalah :

- *Sequence Diagram*

Sequence Diagram merupakan diagram alur yang disusun berdasarkan robustness diagram. Diagram ini dibuat untuk masing-masing use case. Dari semua diagram interaksi UML yang ada, *sequence diagram* adalah diagram yang paling sesuai untuk mengeksplorasi skenario atau arus dari suatu usecase. Tidak hanya itu, diagram ini juga paling mudah untuk digambarkan dan mudah dipahami oleh tim pengembang dan klien.

Manfaat dari pembuat sequence diagram ini yaitu untuk mengidentifikasi pesan apa saja yang dipertukarkan antara objek satu dengan objek lainnya dalam sebuah scenario use case . Sequence diagram digunakan dalam tahapan analisa dan desain [16]. Adapun kegunaan dari sequence diagram ini yaitu untuk mendeskripsikan gambaran dari sebuah sistem ke dalam diagram perilaku (*Use Case Diagram*) yang digambarkan dalam bentuk hubungan antar aktor.

Masing-masing objek termasuk aktor, memiliki *lifeline* vertikal. Pesan yang digambarkan antar objek digambarkan sebagai garis berpanah. *Activation Bar* menunjukkan lamanya eksekusi sebuah proses yang ada, biasanya diawali dengan diterimanya sebuah *message*. Pada Sequence Diagram didalamnya terdapat icon khusus untuk *boundary*, *controller*, dan *entity*. Dibawah ini merupakan penjelasan mengenai masing-masing icon : [18]

**1. Aktor**

Menggambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem

**2. Boundary**

Menggambarkan interaksi antara satu atau lebih *actor* dengan sistem, memodelkan bagian sistem yang bergantung pada pihak lain disekitarnya dan merupakan pembatas sistem dengan dunia luar. Biasanya digunakan untuk menggambarkan GUI atau antarmuka sebuah sistem.

**3. Controller**

Menggambarkan perilaku sistem dan dinamika dari suatu sistem. Icon ini menggambarkan untuk mengontrol alur kerja suatu sistem

**4. Entity**

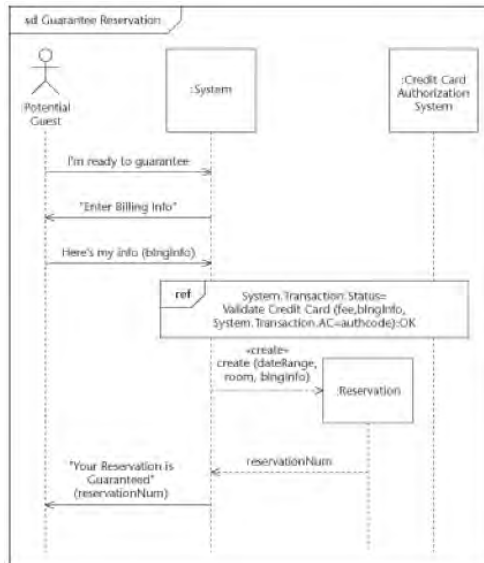
Menggambarkan informasi yang harus disimpan oleh sistem (struktur data dari sebuah sistem). Biasanya digunakan untuk menggambarkan sebuah *database* sistem

**5. Message**

Menggambarkan pesan atau hubungan antar objek yang menunjukkan urutan kejadian yang terjadi

**6. Message to self**

Menggambarkan pesan atau hubungan objek itu sendiri, yang menunjukkan urutan kejadian yang terjadi



**Gambar 2.3 Contoh Sequence Diagram [17]**

- *Milestone 3 : Critical Design review (CDR)*

Peninjauan ulang dilakukan dengan memastikan bahwa desain telah memenuhi semua kebutuhan dari hasil identifikasi sebelumnya. [11]

### ➤ Class Diagram

Class diagram merupakan sebuah diagram yang terdiri dari tiga area pokok utama, yaitu nama, atribut, dan operasi. Nama berfungsi untuk member identitas pada sebuah kelas, atribut berfungsi untuk member karakteristik pada data yang dimiliki suatu objek di dalam kelas, sedangkan operasi berfungsi untuk memberikan sebuah fungsi ke sebuah objek. [19]

Pada class diagram terdiri dari beberapa bagian, yaitu: [19]

- *Class*
- *Interface*



- *Collaborations*
- *Dependency*
- *Generalization*
- *Association Relationship*

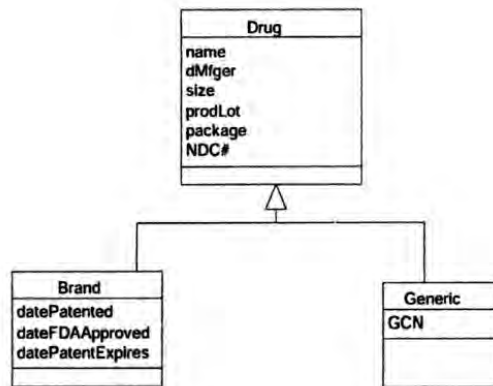
Class diagram memodelkan sumber daya yang essential secara tepat pada sistem yang ingin dibangun. Diagram lainnya mendefinisikan tentang sumber daya seperti nilai atribut, keadaan, batasan pada perilaku yang harus dipetakan kedalam Class Diagram. Class diagram merupakan sumber rujukan dalam mengembangkan kode aplikasi perangkat lunak. [20]

*Class diagram* merupakan grafik yang mengGambar kan koleksi dari deklaratif model dalam sebuah elemen [21] . *Class diagram* menyediakan ringkasan dari sistem target dengan mendeskripsikan objek dan kelas kedalam sistem dan hubungannya diantara keduanya. Class diagram memodelkan sumber daya masing - masing dalam hal struktur, hubungan, dan perilaku. tiga standar kompartemen ditemukan di hampir setiap kelas: nama, atribut, dan operasi.

Hubungan Antar Class dapat dijabarkan sebagai berikut: [22]

- **Asosiasi**, yaitu hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain.
- **Agregasi**, yaitu hubungan yang menyatakan sebuah bagian.
- **Pewarisan**, yaitu hubungan hirarkis antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
- **Hubungan dinamis**, yaitu rangkaian pesan (message) dari satu class kepada class lain.

Berikut merupakan contoh dari Class Diagram yang ditunjukkan pada gambar 2.4 :



Gambar 2.4 Contoh Class Diagram

## 2.2.4 Website

Website adalah sebuah halaman web yang didalamnya terdapat sebuah domain yang mengandung beberapa informasi. Didalam website dapat terdiri dari beberapa halaman web yang dapat berhubungan. Hubungan antara satu dengan halaman web yang lainnya dapat disebut dengan *hyperlink* yang didalamnya terdapat sebuah teks yang digunakan sebagai penghubung yang biasa disebut dengan *hypertext*. Didalam sebuah website terdapat domain yang dapat dimiliki oleh sebuah instansi yang dimana domain tersebut dapat diakses dengan menggunakan internet. Contoh domain antara lain adalah : *yahoo.com* , *google.com* , *detik.com* , *ephi.web.id* dan lain lain. Didalam sebuah website terdapat istilah *homepage*. *Homepage* merupakan sebuah halaman awal pada setiap domain.

Website sendiri terdapat beberapa jenis, lebih dibedakan berdasarkan pada fungsi dan sifat yang digunakan. Website berdasarkan sifatnya, diantaranya adalah : [23]

1. **Website Dinamis** : website yang menyediakan isi yang dapat berubah ubah setiap waktu, lebih mengarah ke website yang berisi berita.

2. **Website Statis** : website yang memiliki konten atau isi yang jarang berubah-ubah. Contohnya adalah website yang berisi profil organisasi.

Website berdasarkan tujuannya, diantaranya adalah :

1. **Personal web** : website yang berisi informasi pribadi seseorang.
2. **Corporate web** : website yang dimiliki oleh sebuah perusahaan.
3. **Portal web** : website yang menyediakan banyak layanan
4. **Forum web** : website yang memiliki tujuan sebagai media diskusi.

### 2.2.5 PHP

PHP merupakan sebuah singkatan dari *Hypertext Preprocessor* yang dimana merupakan suatu bahasa pemrograman yang bersifat *open source*. PHP merupakan sebuah script yang terintegrasi dengan HTML yang terletak pada sebuah server. PHP ini sendiri bersifat dinamis yang dimana dapat digunakan untuk membuat sebuah website yang dinamis juga dengan artian halaman website tersebut dapat ditampilkan pada saat halaman itu dibuka oleh *end user*. Semua script PHP selanjutnya dieksekusi pada server yang dimana script PHP tersebut dijalankan. PHP ini dapat dijalankan secara runtime melalui console yang dapat menjalankan beberapa perintah dari sistem. PHP ini sendiri merupakan bahasa yang paling mudah diaplikasikan karena memiliki referensi yang banyak. PHP juga dilengkapi dengan berbagai macam pendukung lain seperti *Oracle*, *PostgreSQL*, *MySQL* dan lain-lain. [24]

PHP merupakan sebuah bahasa pemrograman yang digunakan untuk membuat halaman web yang bersifat dinamis. Meskipun bahasa PHP ini sering digunakan untuk membangun website, tetapi sebenarnya PHP juga dapat digunakan untuk membuat aplikasi pada GUI. Website yang digunakan dalam membuat PHP memerlukan sebuah software yang bernama *webserver* yang digunakan sebagai tempat pemrosesan kode PHP. Server yang memiliki software PHP akan memproses input berupa

kode PHP yang menghasilkan output halaman website. PHP ini bersifat terbuka dan *multiplatform* yang dimana dapat dijalankan di beberapa web server. [25]

### **2.2.6 Code Igniter**

Codeigniter merupakan framework bahasa pemrograman berbasis dengan PHP yang dirilis pertama kali pada tahun 2006 oleh EllisLab, Inc. Untuk mengembangkan framework ini cukup mudah. Hal ini dikarenakan dalam framework ini, disediakan dokumentasi yang lengkap yang dapat diakses pada website [ellislab.com](http://ellislab.com). Konsep logika dalam Codeigniter ini menggunakan konsep MVC (Model, View, Controller). Berikut ini adalah penjelasan detail dari ketiga bagian tersebut : [26]

1. Model, bagian ini berisikan kode-kode yang digunakan untuk mengakses database
2. View, bagian ini berisikan kode-kode yang berkaitan dengan tampilan, yaitu kode-kode terkait dengan HTML dan PHP yang ditampilkan pada layar browser
3. Controller, berisikan kode-kode yang digunakan untuk menghubungkan antara View dengan Model dan juga sumber daya yang dibutuhkan untuk melakukan proses permintaan layanan HTTP untuk ditampilkan kedalam halaman website.

### 2.2.7 Koperasi



**Gambar 2.6 Logo Koperasi Indonesia [27]**

Koperasi merupakan sekumpulan dari orang-orang yang menjadi satu secara sukarela untuk memenuhi kebutuhan dan aspirasi ekonomi, sosial dan budaya melalui sebuah perusahaan yang dimiliki secara bersama dan dikendalikan secara demokratis. Koperasi merupakan suatu badan kemasyarakatan yang berasal dari luar yang berdasar pada usaha pada masing-masing organisasi atau individu yang sesuai dengan beberapa kegiatan masyarakat Indonesia. Koperasi juga disebut sebagai organisasi masyarakat yang berjalan secara langsung dan dapat saling mengisi aspek kekeluargaan dan gotongroyong.

Dewasa ini, banyak koperasi yang mengesampingkan prinsip koperasi dalam kehidupan sehari-hari karena ingin mendapatkan pengakuan secara langsung dari masyarakat dan dari lingkungan sekitarnya dimana tempat koperasi tersebut beroperasi. Secara umum, koperasi merupakan suatu badan usaha yang dilakukan secara bersama-sama dan bergerak dalam bidang perekonomian yang beranggotakan oleh masyarakat yang memiliki ekonomi yang lemah dan bergabung secara sukarela dan atas dasar kesamaan hak dan kewajiban dalam melakukan usaha yang memiliki tujuan untuk memenuhi kebutuhan pada anggota didalamnya. [28]

Bedasarkan pada UUD nomor 17 tahun 2012 mengenai koperasi dengan adanya UU Koperasi, Koperasi merupakan suatu badan hukum yang didirikan oleh perseorangan atau badan hukum koperasi. Orang-orang yang termasuk dalam anggota suatu koperasi, bertanggungjawab untuk menjalankan usaha yang dapat memenuhi aspirasi dan kebutuhan bersama dalam bidang ekonomi, sosial dan budaya yang sesuai dengan nilai dan prinsip koperasi. Koperasi ini sendiri merupakan perpaduan antara anggota dalam koperasi yang memiliki usaha secara bersama, dan berhubungan dengan suatu badan usaha dengan pasar. [27]

### **2.2.7.1 Koperasi Simpan Pinjam**

Koperasi simpan pinjam merupakan koperasi yang memiliki anggota yang terdiri dari orang-orang yang memiliki kepentingan langsung dalam hal kredit atau simpan pinjam. [29]. Koperasi simpan pinjam dapat disebut juga dengan koperasi kredit yang dimana bergerak dalam usaha pembentukan modal yang menggunakan tabungan-tabungan para anggotanya dengan cara yang murah, cepat dan tepat dengan tujuan untuk produktivitas dan kesejahteraan bersama. [28]

Kegiatan yang terdapat dalam koperasi simpan pinjam dijalankan sepenuhnya oleh orang-orang yang biasa disebut dengan *unit simpan pinjam*. Yang dimaksud unit disini adalah unit yang bergerak dibidang usaha simpan pinjam yang merupakan bagian dari kegiatan usaha pada koperasi yang bersangkutan. [29]

Adapun peranan dari koperasi simpan pinjam ini yaitu turut mengembangkan perekonomian di Indonesia terutama bagi anggota yang lain. Berikut merupakan beberapa peranan dari koperasi simpan pinjam di Indonesia : [28]

1. Mendidik anggota koperasi agar lebih sering menabung secara teratur sehingga anggota dapat memiliki modal tersendiri
2. Membantu keperluan kredit anggota dengan mengikuti beberapa persyaratan
3. Anggota dapat menambah pengetahuan mengenai koperasi
4. Dapat memperoleh pinjaman dengan mudah dan aman
5. Terdapat jaminan pada saat meminjam, sehingga aman

Perbedaan koperasi simpan pinjam dengan koperasi simpan pinjam adalah, koperasi simpan pinjam memberikan modal bagi anggotanya yang melakukan simpanan atau pinjaman dari hasil usaha yang ada termasuk cadangan modal. Sumber dari simpanan tersebut bervariasi, diantaranya adalah : [28]

1. **Simpanan Pokok**, yaitu simpanan yang diberikan kepada anggota pada awal dilakukan setoran dan menjadi simpanan dalam bentuk yang permanen.
2. **Simpanan Wajib**, yaitu simpanan yang dapat diambil kapan saja dalam jangka waktu tertentu
3. **Simpanan Sukarela**, adalah simpanan yang diterima oleh anggota bukan dari anggota koperasi itu sendiri.

Adapun manfaat dari koperasi simpan pinjam ini sendiri adalah : [28]

1. Anggota yang melakukan proses simpan pinjam mendapatkan modal untuk dapat digunakan dalam mengembangkan usaha atau berwiraswasta
2. Anggota dapat terbantu karena adanya koperasi ini karena dapat digunakan untuk biaya sekolah, rumah sakit, dan lain-lain

## 2.2.8 Profil Perusahaan PT Wonosari Jaya

Sejarah berdirinya suatu koperasi karyawan tidak bisa lepas dari perusahaan dimana karyawan tersebut bekerja. PT. Wonosari Jaya Surabaya yang bergerak di Industri Kawat Baja yang

merupakan ladang penghidupan karyawannya, sudah berdiri sejak tahun 1972 yang awal pendiriannya merupakan Industri Kawat Baja terbesar di ASEAN. Seiring dengan perkembangan jaman dan tuntutan akan adanya serikat dan berkumpul para karyawan dalam kegiatan positif sehingga pada tanggal 30 September 1982 terbentuklah salah satu organisasi sosial berazaskan kekeluargaan dengan semangat kegotongroyongan, yaitu Koperasi Karyawan PT. WONOSARI JAYA sesuai Badan Hukum No. 5341/BH/II/82. Dalam perkembangannya Koperasi Karyawan selalu berusaha mengupdate hal-hal baru agar tujuan organisasi tercapai sesuai harapan.

Kegiatan Pokok Koperasi identik dengan Simpan Pinjam, begitu juga dengan Kopkar Pt. Wonosari Jaya. Awal pendirian sampai 15 tahun terbentuknya, kegiatan pokok yang memberi kontribusi terbesar dalam menghasilkan Sisa Hasil Usaha adalah dari aktivitas anggota berupa Pinjaman Reguler. Selanjutnya dalam perkembangannya usaha Kopkar PT. Wonosari Jaya membuka beberapa jenis aktivitas untuk meningkatkan Laba usahanya selain Pinjaman Reguler dan juga menghimpun dana yang cukup baik dari intern maupun ekstern dalam kaitan untuk mencapai tujuan organisasi, diantaranya:



- a. Pinjaman Barang/Non Reguler/khusus.
- b. Pinjaman Kredit Paket Sekolah.
- c. Simpanan Sukarela
- d. Simpanan Berjangka Koperasi
- e. Arisan.
- f. Sembako.
- g. Menjual Tanah Kavling.

Hal-hal yang dilakukan oleh masing-masing Komponen dalam kegiatan Pinjam:

a. Anggota Koperasi:

- Mengisi Formulir pengajuan Pinjaman.
- Memilih Pinjaman yang akan diambil (Pinjaman Reguler/Non Reguler)
- Mengisi jumlah uang yang akan dipinjam.
- Mengisi jangka waktu angsuran.
- Mencatat apakah ada Pelunasan angsuran yang dipercepat atau tidak dalam kaitan pengajuan pinjaman ini.

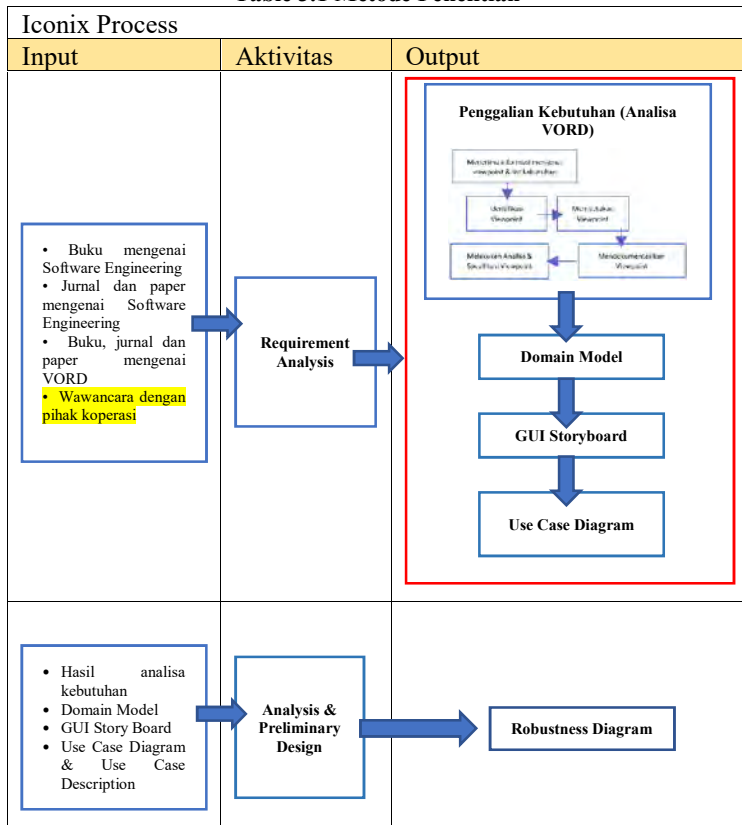
b. Divisi Simpan Pinjam

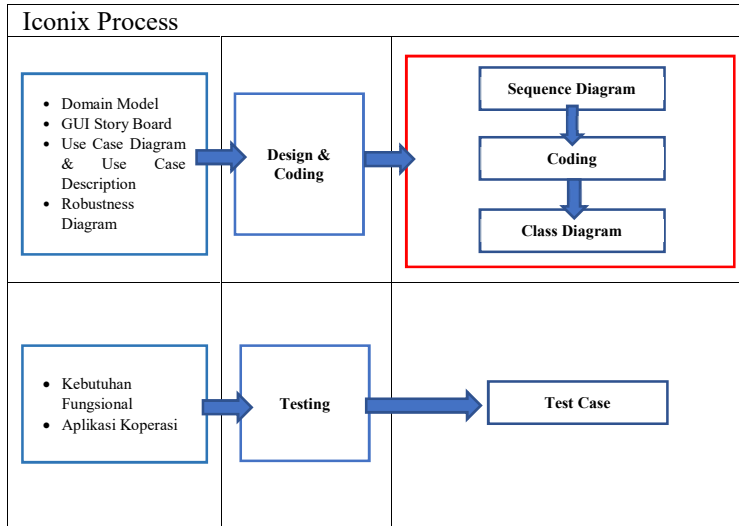
- Memeriksa kebenaran data atas formulir pengajuan pinjaman.
- Hal pokok yang diperiksa divisi simpan pinjam atas formulir tersebut adalah besarnya pinjaman, besarnya angsuran perbulan dan Total Angsuran dari anggota tersebut dalam satu bulan melebihi dari 60% gaji per bulan, atau tidak.
- Bila sudah benar maka Formulir divalidasi dan dimintakan persetujuan kepada Ketua Koperasi dan selanjutnya dibuatkan rekap pinjaman sebagai dasar untuk tanda terima pencairan uang di Bendahara dan penyerahan uang ke anggota. Kebijakan yang digunakan adalah “Total angusran pinjaman per bulan tidak boleh lebih dari 60% Gaji per bulan”

### BAB III METODOLOGI PENELITIAN

Pada bagian ini akan dijelaskan mengenai metodologi dalam melakukan pengerjaan Tugas Akhir, sehingga langkah-langkah pengerjaan menjadi lebih sistematis dan terorganisir lebih rapi. Berikut ini merupakan tahapan metodologi pengerjaan tugas akhir :

**Table 3.1 Metode Penelitian**





### 3.1 Requirement Analysis

#### 3.1.1 Penggalan Kebutuhan

##### a. Wawancara

Tahapan pertama yang dilakukan adalah wawancara. Wawancara ini dilakukan sebagai upaya mengidentifikasi permasalahan dari perangkat lunak yang akan dibuat. Wawancara merupakan hal yang penting dalam tahap penggalan kebutuhan. Apabila saat wawancara ditemukan data yang kurang lengkap atau tidak sesuai dengan kebutuhan pengguna, maka harus ada upaya perbaikan terhadap data yang telah dikumpulkan sebelumnya. Persiapan wawancara yang tidak matang menyebabkan pengumpulan data tidak maksimal karena data yang tidak sesuai tersebut gagal dikoreksi sedangkan data yang kurang lengkap gagal dilengkapi. Akibatnya tahap analisa kebutuhan pengguna terhambat.

Sehingga didapatkan spesifikasi kebutuhan perangkat lunak yaitu Kebutuhan Fungsional dan Kebutuhan Non-fungsional.

Hasil wawancara yang didapat akan digunakan sebagai masukan dalam analisa kebutuhan.

## **b. Analisis**

Setelah pada tahap pengumpulan data terkumpul, analisa kebutuhan dapat mulai dilakukan. Pada tahap ini, peneliti menggunakan metode VORD dalam tahapan analisa kebutuhan. Setelah dilakukan wawancara, tahapan selanjutnya adalah analisa dari hasil wawancara yang dilakukan dan dilakukan review dokumen, dimana dokumen yang ditinjau hanya terdiri dari dokumen SOP dan alur kerja koperasi simpan pinjam. Aktifitas yang dilakukan adalah membaca prosedur yang telah ditulis pada dokumen SOP serta mereview analisa kebutuhan pengguna koperasi simpan pinjam PT Wonosari Jaya, yang dimana informasi yang didapatkan dari dokumen ini salah satunya untuk mengetahui sistem informasi manajemen apa saja yang dibutuhkan aktor dalam menjalankan aplikasi koperasi simpan pinjam selanjutnya.

Metode yang digunakan untuk melakukan proses analisa kebutuhan adalah metode VORD (*Viewpoint Oriented Requirement Definition*). Metode VORD menggambarkan kebutuhan sistem sebagai suatu layanan yang diberikan oleh sistem terhadap viewpoint. VORD didasarkan pada tiga tahap utama, antara lain:

### **a. Viewpoint Identification**

Identifikasi viewpoint meliputi penemuan viewpoint berdasarkan layanan yang akan diterima oleh setiap viewpoint.

### **b. Viewpoint Documentation**

Dokumentasi viewpoint meliputi cara untuk mendeskripsikan setiap viewpoint dan layanan yang telah ditentukan sebelumnya.

### c. Viewpoint Analyze & Specification

Tahap ini merupakan tahap akhir yang meliputi analisa kebutuhan pada setiap viewpoint yang telah didokumentasikan, setelah itu dilakukan spesifikasi dengan menggunakan *matrix* untuk menghubungkan setiap viewpoint dengan kebutuhan yang telah didokumentasikan sebelumnya.

Setelah proses spesifikasi kebutuhan selesai, selanjutnya akan dilakukan evaluasi dokumen yaitu dengan mencocokkan apakah *viewpoint (pengguna)* dan kebutuhan yang didefinisikan telah cocok atau belum.

Produk yang dihasilkan dari tahap identifikasi kebutuhan ini adalah dokumen spesifikasi kebutuhan. Dari metode ini dapat diketahui siapa saja pengguna yang terlibat pada kebutuhan-kebutuhan yang telah dijabarkan dan yang diperlukan oleh pengguna pada sistem informasi koperasi simpan pinjam.

Setelah analisa kebutuhan dilakukan, tahap selanjutnya adalah menggambarkan kebutuhan pengguna yang telah didapatkan selanjutnya dijabarkan dalam bentuk *Use Case Diagram*, dan *Use Case Deskripsi*. Didalamnya berisi gambaran umum beserta penjelasan mengenai proses bisnis pada koperasi karyawan PT Wonosari Jaya. Penjabaran kebutuhan yang digambarkan dalam *Use Case Diagram* dan *Use Case Deskripsi* merupakan tahapan awal yang sangat penting karena berpengaruh besar terhadap tahapan-tahapan selanjutnya.

### 3.1.2 Domain Model

Domain modeling merupakan pondasi awal pada struktur statik pada UML. Domain modeling didapatkan dari mengekstrak kata benda yang didapatkan dari functional requirements. Kata benda yang didapatkan saling dihubungkan sesuai kebutuhan dari perangkat lunak.

Pada tahapan ini selanjutnya akan dibuat domain model berdasarkan analisa kebutuhan yang telah didefinisikan sebelumnya.

### **3.1.3 GUI Storyboard**

GUI Storyboard disesuaikan dengan functional requirements yang telah didefinisikan sebelumnya. Dari functional requirement tersebut, kemudian peneliti merancang GUI untuk Sistem Informasi Koperasi yang akan dibangun nantinya.

### **3.1.4 Use Case Diagram**

Pada tahapan ini selanjutnya akan dibuat use case berdasarkan Analisa Kebutuhan, Domain Model, dan GUI storyboard yang telah didefinisikan sebelumnya.

### **3.1.5 Use Case Description**

Pada tahapan ini selanjutnya akan dibuat use case description berdasarkan Analisa Kebutuhan, Domain Model, GUI Storyboard dan Use Case Diagram yang telah didefinisikan sebelumnya.

## **3.2 Analysis and Preliminary Design**

### **3.2.1 Robustness Diagram**

Pada tahapan ini selanjutnya akan dibuat use case description berdasarkan Analisa Kebutuhan, Domain Model, GUI Storyboard, Use Case Diagram dan Use Case Description yang telah didefinisikan sebelumnya.

### **3.3 Design & Coding**

#### **3.3.1. Sequence Diagram**

Pada tahapan ini akan dijelaskan mengenai sequence diagram pada aplikasi koperasi PT Wonosari Jaya yang berdasar pada Ucase Diagram dan Usecase Description.

#### **3.3.2. Class Diagram**

Pada tahapan ini akan dijelaskan mengenai class diagram yang berdasar pada Sequence Diagram yang telah didefinisikan sebelumnya.

### **3.4 Testing**

Pada tahapan ini akan dijelaskan mengenai pengujian aplikasi dengan menggunakan *Test Case* dengan input yang digunakan adalah kebutuhan fungsional dan aplikasi koperasi PT Wonosari Jaya. Pengujian dengan menggunakan metode ini dilakukan dengan cara menguji skenario utama dan skenario alternative pada masing-masing use case.

## **BAB IV PERANCANGAN**

Pada bagian ini, akan dijelaskan mengenai tahapan pembangunan sistem yang dimulai dari tahap wawancara hingga desain class diagram.

### **4.1 Requirement Analysis**

Pada tahap pembangunan sistem informasi koperasi, tahapan pertama yang dilakukan adalah *Requirement Analysis*. Proses *requirement analysis* terdiri dari beberapa metode, diantaranya wawancara, analisa kebutuhan dengan menggunakan metode VORD, Domain Model dan Use Case.

#### **4.1.1 Penggalan Kebutuhan**

##### **4.1.1.1 Wawancara**

Sebelum membangun aplikasi, peneliti melakukan wawancara dengan ketua koperasi simpan pinjam PT Wonosari Jaya. Wawancara merupakan salah satu *tools* yang digunakan pada proses penggalan kebutuhan. Sebelum melakukan wawancara, peneliti melakukan tahapan-tahapan sebagai berikut :

**a. Melakukan perjanjian dengan pihak *Interviewee***

Pada tahap ini, peneliti sebelumnya telah membuat perjanjian pertemuan yang dimana membahas mengenai informasi proses bisnis pada koperasi simpan pinjam serta mengenai kebutuhan apa saja yang dibutuhkan pada aplikasi.

**b. Mendesain pertanyaan-pertanyaan untuk wawancara**

Setelah melakukan perjanjian dengan pihak *interviewee*, tahap selanjutnya adalah membuat list pertanyaan yang selanjutnya akan digunakan untuk wawancara. Tipe wawancara yang digunakan adalah



semi-terstruktur (*semi-structured interviews*). Tipe ini adalah jenis interview yang menghubungkan pertanyaan awal yang telah dibuat dengan pertanyaan spontanitas yang dapat mengetahui lebih dalam ketertarikan dari *interviewee*. [30]

**c. Persiapan wawancara**

Pada tahap ini berisi apa saja yang dibutuhkan pada saat wawancara. Didalamnya terdiri dari jadwal dan tempat wawancara yang telah didokumentasikan pada bagian lampiran.

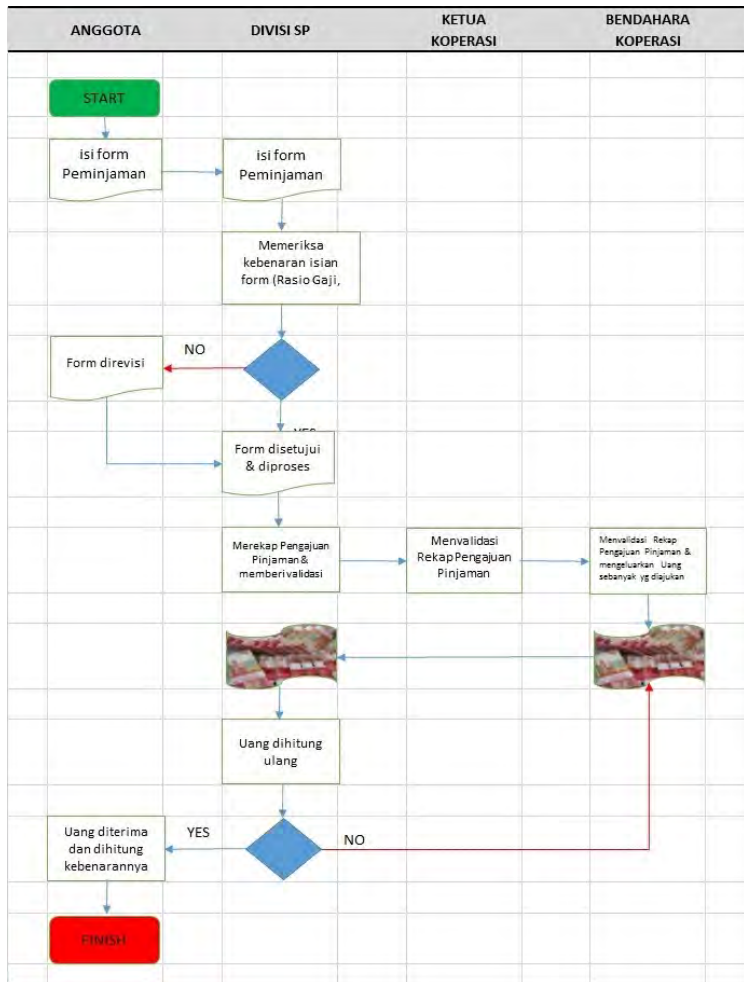
**d. Melakukan wawancara**

Tahap ini merupakan tahapan terakhir pada proses wawancara, yaitu peneliti mulai melakukan wawancara dengan *interviewee*. Berikut merupakan daftar pertanyaan yang diajukan terkait pembuatan aplikasi koperasi karyawan PT Wonosari Jaya :

1. Bagaimana awal berdirinya perusahaan PT Wonosari Jaya ini?
2. Bagaimana struktur organisasi yang ada pada koperasi perusahaan PT Wonosari Jaya?
3. Bagaimana proses bisnis yang ada pada koperasi PT Wonosari Jaya ?
4. Bagaimana proses simpanan pada koperasi perusahaan PT Wonosari Jaya?
5. Bagaimana proses pinjaman pada koperasi perusahaan PT Wonosari Jaya?
6. Apakah terdapat kendala/permasalahan yang ada pada koperasi PT Wonosari Jaya? Jika iya, apa saja kendala yang ada?
7. Apakah harapan dari bapak terkait dengan proses transaksi pada koperasi simpan pinjam PT Wonosari Jaya pada perangkat lunak ini?

Salah satu hasil dari pertanyaan wawancara yang didapatkan adalah proses bisnis pada perusahaan, berikut

merupakan proses bisnis pada transaksi simpanan dan transaksi pinjaman :



**Gambar 4.1** Proses Bisnis Transaksi Pinjaman



Pada tahapan ini, peneliti melakukan translasi wawancara kedalam cerita pengguna, membuat daftar kebutuhan aplikasi berdasarkan hasil translasi wawancara kedalam cerita pengguna, membuat daftar kebutuhan aplikasi, mendeskripsikan perangkat lunak yang akan dibuat dan melakukan klasifikasi pengguna dari perangkat lunak yang akan dibuat. Dengan adanya cerita pengguna ini, akan dihasilkan sebuah kebutuhan/Requirement potensial/Fitur potensial.

**Tabel 4.1 User Story**

<b>User Story Code</b>	<b>User Story Title</b>	<b>Description</b>	<b>Requirement</b>
US-01-01	Aplikasi mudah diakses dimanapun	Aplikasi yang akan dibuat nanti harus mudah diakses oleh pengguna dari aplikasi ini, sehingga tidak harus datang ke kantor untuk menjalankan aplikasi ini	1. Aplikasi berbasis web
US-01-02	Memudahkan proses pendaftaran bagi karyawan	Karyawan dapat melakukan pendaftaran hanya melalui aplikasi, tanpa harus datang ke kantor.	1. Masing-masing karyawan memiliki satu username dan password yang dapat digunakan dalam melakukan transaksi 2. Bagian pendaftaran mencatat data-data yang diperlukan diantaranya : Nama, NIK, Alamat, Nomor Telepon, Jabatan

User Story Code	User Story Title	Description	Requirement
US-01-03	Aplikasi Koperasi Simpan Pinjam nanti hanya bisa digunakan oleh orang-orang tertentu	Aplikasi hanya dapat digunakan oleh orang-orang yang terlibat dalam perangkat lunak koperasi simpan pinjam, sesuai dengan proses bisnis yang ada di Perusahaan PT Wonosari Jaya ini. Selain itu, mereka tidak boleh mengakses. Hanya dapat melakukan akses jika mendapatkan izin dari admin(Divisi SP).	Aplikasi yang terdapat fitur: 1. Login/Logout 2. Fitur Ubah Password
US-01-04	Pengubahan data dari proses registrasi anggota	Divisi SP mempunyai kuasa dalam melakukan perubahan/customisasi data registrasi transaksi	1. Fitur perubahan data
US-01-05	Aplikasi koperasi diharapkan dapat merekam data-data transaksi	Aplikasi koperasi diharapkan dapat merekam data transaksi dengan cepat dan langsung terhubung ke tahapan berikutnya	Sistem merekam data-data transaksi yang dilakukan oleh karyawan dan langsung menghubungkan ke proses berikutnya
US-01-06	Adanya fitur tambah pengguna	Aplikasi koperasi diharapkan dapat digunakan admin (Divisi SP) dalam menambahkan pengguna	1. Admin membuat <i>username</i> dan <i>password</i> untuk karyawan yang digunakan untuk login

User Story Code	User Story Title	Description	Requirement
US-01-07	Verifikasi dan validasi data-data transaksi	Aplikasi koperasi diharapkan dapat melihat data transaksi, kemudian Divisi SP, Ketua, dan Bendahara melakukan verifikasi data yang telah diisikan oleh karyawan	<ol style="list-style-type: none"> <li>1. Dapat melihat data transaksi</li> <li>2. Divisi SP, Ketua, dan Bendahara dapat melakukan verifikasi data transaksi</li> </ol>
US-01-08	Melihat rekap data	Pengguna aplikasi dapat melihat data transaksi	<ol style="list-style-type: none"> <li>1. Fitur rekap data transaksi</li> </ol>
US-01-09	Mencetak dokumen laporan	Setelah Admin melakukan validasi, Divisi SP dapat melakukan pencetakan dokumen hasil transaksi yang selanjutnya diserahkan kepada karyawan	<ol style="list-style-type: none"> <li>1. Fitur cetak dokumen yang dapat diunduh melalui pdf</li> </ol>
US-01-10	Pengubahan data hanya dapat dilakukan oleh Divisi SP	Apabila terdapat data yang tidak sesuai, maka Divisi SP dapat menggunakan fitur <i>edit</i> untuk merubah data anggota	<ol style="list-style-type: none"> <li>1. Fitur edit data yang terletak disebelah kanan tabel data anggota</li> </ol>

#### 4.1.1.3 Membuat Daftar Kebutuhan Aplikasi

Daftar kebutuhan aplikasi dibuat berdasarkan hasil analisa terhadap cerita pengguna (User Story). Analisa kebutuhan pada system informasi koperasi ini peneliti menggunakan metode VORD (*Viewpoint Oriented Requirement Definition*). Daftar

kebutuhan aplikasi ini nantinya akan dijadikan acuan dalam pembuatan dokumen Spesifikasi Kebutuhan Perangkat Lunak (SKPL). Sebelumnya, telah didefinisikan tugas dan fungsi dari masing-masing aktor yang didapat dari dokumen perusahaan, diantaranya adalah :

1. Anggota koperasi yang hanya dapat melakukan tambah transaksi, melihat dan mencari data transaksi.
2. Divisi SP memiliki peranan penting dalam koperasi karyawan PT Wonosari Jaya, diantaranya adalah membuat akun pengguna, melakukan verifikasi transaksi, melakukan unduh data, melakukan display dan cari data transaksi serta edit data transaksi
3. Ketua Koperasi memiliki tugas dalam verifikasi data transaksi, serta melakukan display dan cari data transaksi serta edit data transaksi
4. Bendahara, dapat melakukan verifikasi data transaksi, update data simpanan wajib, unduh data transaksi serta display dan cari data transaksi serta edit data transaksi.

Pada Tabel 4.2 merupakan daftar kebutuhan yang ada pada aplikasi koperasi simpan pinjam yang sudah dikonfirmasi kepada client.

## KEBUTUHAN FUNGSIONAL

**Tabel 4.2 Kebutuhan Fungsional**

<b>Kode</b>	<b>Penjelasan</b>
KF01	Fitur Autentifikasi
KF02	Terdapat fitur display data dan cari data yang dilakukan oleh semua pengguna
KF03	Terdapat fitur pengubahan data yang dilakukan oleh Divisi SP
KF04	Terdapat fitur verifikasi yang hanya dapat dilakukan oleh Divisi SP
KF05	Terdapat fitur display data dan cari data yang hanya dapat dilakukan oleh Divisi SP
KF06	Terdapat fitur unduh yang hanya dapat digunakan oleh Bendahara dan Divisi SP

Kode	Penjelasan
KF07	Terdapat fitur display data, cari data dan verifikasi data yang hanya digunakan oleh Divisi SP, Ketua dan Bendahara
KF08	Terdapat fitur cari data yang hanya dapat digunakan oleh Bendahara
KF09	Terdapat fitur update data yang hanya dapat digunakan oleh Bendahara
KF10	Terdapat fitur tambah data yang hanya dapat dilakukan oleh anggota
KF11	Terdapat fitur tambah data yang hanya dapat dilakukan oleh divisi SP

#### a. Viewpoint Identification

Identifikasi *viewpoint* meliputi peneuan *viewpoint* berdasarkan layanan yang akan diterima oleh setiap *viewpoint*. Pada tahap ini, dilakukan identifikasi terhadap kebutuhan yang dijelaskan dengan mendaftar semua *stakeholder* yang terdapat dalam sistem tersebut dengan melakukan *brainstorming*.

Tujuan analisa kebutuhan ini adalah untuk membagi ruang lingkup sistem berdasarkan kebutuhan fungsional untuk masing-masing pengguna dan kebutuhan non-fungsional yang berkaitan dengan kebutuhan fungsional tersebut

Viewpoint yang terdapat didalam sistem :

1. Anggota
2. Divisi SP
3. Ketua koperasi
4. Bendahara koperasi
5. Peraturan Koperasi
6. Sistem Informasi Koperasi



Sesuai dengan permasalahan yang telah diidentifikasi sebelumnya, dapat diketahui layanan apa saja yang dapat diberikan sistem kepada pengguna, diantaranya adalah :

**Tabel 4.3 Pendefinisian Requirement**

<b>Requirement</b>	<b>Description</b>
R1	Pengguna dapat melakukan fitur <i>login</i> dan <i>logout</i>
R2	Pengguna dapat menggunakan fitur display data untuk melihat data-data transaksi yang telah tersimpan
R3	Pengguna dapat menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi
R4	Divisi SP dapat menggunakan fitur edit setelah anggota melakukan transaksi. Divisi SP dapat melakukan pengubahan data dengan menggunakan fitur edit disebelah kanan tabel transaksi. Divisi SP hanya dapat mengedit nama data anggota, data simpanan, data tarikan, dan data pinjaman.
R5	Divisi SP memiliki kewenangan untuk memverifikasi data simpanan setelah anggota melakukan tambah simpanan
R6	Data pelunasan pinjaman hanya dapat diakses oleh Divisi SP. Divisi SP memiliki kewenangan untuk melihat data pelunasan pinjaman.
R7	Didalam data pelunasan peminjaman, terdapat fitur cari data yang digunakan untuk memudahkan Divisi SP dalam mencari data transaksi.
R8	Bendahara memiliki kewenangan untuk menggunakan fitur unduh data. Fitur tersebut digunakan oleh Bendahara diteruskan ke tahap selanjutnya. Data yang telah diunduh selanjutnya dikirimkan ke Ketua untuk di tanda tangani.
R9	Divisi SP memiliki kewenangan untuk menggunakan fitur unduh data. Fitur tersebut digunakan oleh Divisi SP untuk

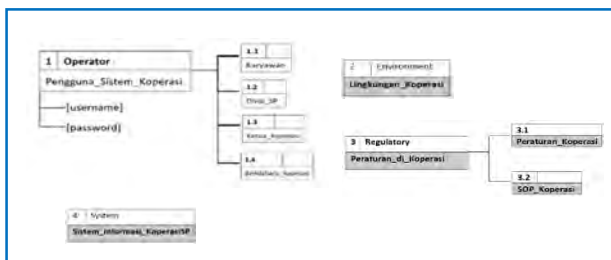
Requirement	Description
	diteruskan ke tahap selanjutnya. Data yang telah diunduh selanjutnya dikirimkan ke Ketua untuk di tanda tangani.
R10	Divisi SP, Bendahara dan Ketua dapat menggunakan fitur display data untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi. Selanjutnya, Divisi SP, Ketuadan Bendahara dapat melakukan verifikasi data apabila data yang diajukan oleh anggota sudah benar.
R11	Bendahara memiliki tugas untuk mengatur data simpanan wajib anggota. Oleh karena itu, Bendahara memiliki satu fitur yaitu “Cari Data” yang memudahkan Bendahara dalam mencari data simpanan wajib anggota.
R12	Bendahara memiliki tugas untuk mengatur data simpanan wajib anggota. Hanya bendahara yang memiliki kewenangan untuk mengupdate data simpanan wajib anggota. Data simpanan wajib anggota bersifat bulanan.
R13	Anggota memiliki peran penting dalam system informasi koperasi. Oleh karena itu, anggota memiliki fitur tambah data yang dapat digunakan untuk menambah data simpanan, data tarikan dan data pinjaman. Fitur tambah data hanya dapat digunakan ketika anggota ingin melakukan transaksi di koperasi.
R14	Divisi SP memiliki peran utama dalam system informasi koperasi. Karena, sebagian besar data dikelola oleh Divisi SP. Oleh karena itu, Divisi SP memiliki kewenangan untuk menambah data anggota

Requirement	Description
	yang didalamnya terdiri dari Nama Anggota, username dan password (digunakan anggota untuk melakukan transaksi). Fitur tambah data yang digunakan oleh Divisi SP adalah pada fitur tambah data anggota dan tambah data angsuran.

## b. Viewpoint Structuring

Strukturisasi viewpoint digunakan untuk memberikan gambaran mengenai pengguna dari sistem informasi koperasi, seperti yang ditunjukkan pada gambar 4.1 Pengguna Sistem Informasi Koperasi Simpan Pinjam ini memiliki empat aktor yang terdiri dari karyawan, Divisi SP, Ketua Koperasi dan Bendahara koperasi. Masing-masing aktor memiliki tugas yang berbeda-beda dan lebih sesuai dengan hak aksesnya.

Strukturisasi *viewpoint* dilakukan untuk menggambarkan siapa saja yang menjadi pengguna dalam Sistem Informasi Koperasi Simpan Pinjam. Ilustrasi strukturisasi *viewpoint* pengguna Sistem Informasi Koperasi Simpan Pinjam dapat dilihat pada gambar 4.1 berikut ini :



Gambar 4.3 Viewpoint Structuring

Berdasarkan viewpoint diatas, dapat dibuat diskripsi dari masing-masing viewpoint yaitu sebagai berikut:

**Tabel 4.4 Deskripsi Viewpoint**

Identifier	Label	Type	Description
1.	Pengguna Sistem Koperasi	Operator/ Pengguna	Viewpoint ini merupakan pengguna yang mengoperasikan sistem yang dibagi sesuai dengan hak aksesnya. Pengguna ini meliputi karyawan, Divisi SP, ketua koperasi dan Bendahara koperasi
1.1	Anggota	Operator/ Pengguna	View point ini merupakan aktor yang terlibat langsung dalam sistem. Dimana karyawan ini akan membutuhkan informasi mengenai koperasi dan dapat mengakses sesuai dengan hak aksesnya sebagai karyawan. Selain itu karyawan ini dapat melakukan registrasi terlebih dahulu sebelum melakukan login, dan juga dapat memilih ingin melakukan simpanan, penarikan uang atau pinjaman.

Identifier	Label	Type	Description
1.2	Divisi Simpan Pinjam	Operator/ Pengguna	View point ini bertanggung jawab dalam mengelola, mengecek dan memvalidasi form pendaftaran karyawan, dan meneruskan form tersebut kepada ketua koperasi. Divisi SP ini juga dapat mengakses sistem informasi koperasi simpan pinjam sesuai dengan hak aksesnya.
1.3	Ketua Koperasi	Operator/ Pengguna	View point ini bertanggung jawab untuk memvalidasi dan menyetujui form pendaftaran karyawan yang telah diterima dari pihak Divisi SP dan meneruskan form tersebut kepada Bendahara koperasi. Ketua koperasi ini juga dapat mengakses sistem informasi koperasi simpan pinjam sesuai dengan hak aksesnya.
1.4	Bendahara Koperasi	Operator/ Pengguna	View point ini bertanggung jawab untuk menyetujui dan mencetak bukti validasi form yang telah diterima dari ketua koperasi. Bendahara koperasi

Identifier	Label	Type	Description
			ini juga dapat mengakses sistem informasi koperasi simpan pinjam sesuai dengan hak aksesnya.
2.	Lingkungan Koperasi	Environment	Viewpoint ini merupakan sebuah lingkungan yang selanjutnya akan menggunakan sistem informasi koperasi.
3.	Peraturan Koperasi	Regulatory	View point ini berisikan mengenai peraturan-peraturan apa saja yang terdapat pada koperasi simpan pinjam PT Wonosari Jaya

### c. Viewpoint Documentation

Berdasarkan ruang lingkup sistem yang telah ditentukan, dilakukan analisa kebutuhan dengan menggunakan metode VORD (Viewpoint-Oriented Requirements Definition). Dokumentasi *viewpoint* meliputi cara untuk mendeskripsikan setiap viewpoint dan layanan yang telah ditentukan sebelumnya. Tujuan dari *viewpoint documentation* ini adalah untuk memetakan kebutuhan sistem atau layanan terhadap masing-masing *viewpoint* sesuai dengan *viewpoint structuring*. Pemetaan kebutuhan tersebut meliputi kebutuhan fungsional dan non-fungsional sistem. Hasil dari *viewpoint documentation* dapat digunakan sebagai acuan dalam pembuatan *use case*. Bentuk dari dokumentasi VORD seperti ditunjukkan pada tabel dibawah ini :

Dari dokumentasi viewpoint diatas, didapatkan identifikasi direct dan indirect viewpoint yang akan dijelaskan pada tabel berikut ini:

➤ **Kebutuhan Direct Viewpoint**

**Tabel 4.5 Direct Viewpoint**

<b>Viewpoint</b>		<b>Requirement</b>			
<b>Identifier</b>	<b>Label</b>	<b>Description</b>		<b>Type</b>	<b>Souce VP</b>
1.	Pengguna Sistem Informasi Koperasi	<b>1.1</b>	<b>Tersedia fungsi untuk masuk ke dalam sistem (login) dan keluar dari system (logout)</b>	Fun	1
		1.1.1	Fungsi login menggunakan parameter username dan password	Nfun	
		<b>1.2</b>	<b>Melihat informasi koperasi</b>	Fun	
		1.2.1	Tersedia fungsi untuk melihat data simpanan	Fun	
		1.2.2	Tersedia fungsi untuk melihat data tarikan	Fun	
		1.2.3	Tersedia fungsi untuk melihat data pinjaman	Fun	
		1.2.4	Tersedia fungsi untuk melihat data angsuran	Fun	

Viewpoint		Requirement			
		1.2.5	Data transaksi koperasi diurutkan berdasarkan transaksi yang paling terbaru	Nfun	
		<b>1.3</b>	<b>Mencari informasi data transaksi</b>	Fun	
		1.3.1	Tersedia fungsi untuk mencari data simpanan	Fun	
		1.3.2	Tersedia fungsi untuk mencari data pinjaman	Fun	
		1.3.3	Tersedia fungsi untuk mencari data tarikan	Fun	
		1.3.4	Tersedia fungsi untuk mencari data angsuran	Fun	
		1.3.5	Terdapat 4 kategori yang ingin dicari serta kolom keyword yang digunakan untuk mencari data	Nfun	
1.1	Anggota	<b>1.1.1</b>	<b>Terdapat fungsi tambah data simpanan</b>	<b>Fun</b>	1.1
		1.1.1.1	Form tambah transaksi simpanan terdiri dari nomor anggota, tanggal, nominal yang akan disimpan, serta jenis simpanan	Nfun	



Viewpoint		Requirement			
		<b>1.1.2</b>	<b>Terdapat fungsi tambah data tarikan</b>	<b>Fun</b>	
		1.1.2.1	Form tambah transaksi tarikan terdiri dari nomor anggota, tanggal, sisa saldo, dan nominal tarikan	Nfun	
		<b>1.1.3</b>	<b>Terdapat fungsi tambah data pinjaman</b>	<b>Fun</b>	
		1.1.3.1	Form tambah transaksi pinjaman terdiri dari nomor anggota, tanggal, nominal pinjaman yang ingin diajukan, jenis pinjaman, jenis angsuran, total angsuran per bulan, dan rasio	<b>Nfun</b>	
		1.1.3.1.1	Fitur tambah pinjaman tidak dapat digunakan apabila anggota belum membayar setidaknya 50% dari pinjaman sebelumnya	Nfun	
		<b>1.1.4</b>	<b>Memilih opsi simpanan, tarikan, pinjaman dan angsuran</b>	Nfun	
		<b>1.1.5</b>	<b>Terdapat fungsi menampilkan data simpanan wajib</b>	Fun	

Viewpoint		Requirement			
1.3	Ketua Koperasi	1.3.1	Memilih opsi simpanan, tarikan, pinjaman dan angsuran	NFun	1.3
		1.3.2	Tersedia fitur Display Data Anggota	Fun	
		1.3.3	Tersedia fitur verifikasi form data tarikan yang telah didapat dari Bendahara	Fun	
		1.3.4	Tersedia fitur verifikasi form data pinjaman yang telah didapat dari Bendahara	Fun	
		1.3.5	Tersedia fitur Cari Data Anggota	Fun	
1.4	Divisi SP	1.4.1	Memilih opsi simpanan, tarikan, pinjaman dan angsuran	NFun	1.4
		1.4.2	Tersedia fitur Display Data Anggota	Fun	
		1.4.3	Tersedia fitur verifikasi form data tarikan	Fun	
		1.4.4	Tersedia fitur verifikasi form data pinjaman	Fun	
		1.4.5	Tersedia fitur Cari Data Anggota	Fun	
		1.4.6	Tersedia fungsi edit data anggota	Fun	

Viewpoint		Requirement			
		1.4.7	Tersedia fungsi edit data simpanan	Fun	
		1.4.8	Tersedia fungsi edit data tarikan	Fun	
		1.4.9	Tersedia fungsi edit data pinjaman	Fun	
		1.4.10	Tersedia fungsi verifikasi data simpanan	Fun	
		1.4.10.1	Fitur verifikasi data simpanan hanya dapat dilakukan oleh Divisi SP	Nfun	
		1.4.11	Tersedia fungsi display data pelunasan pinjaman	Fun	
		1.4.12	Tersedia fungsi cari data pelunasan pinjaman	Fun	
		1.4.13	Tersedia fungsi unduh data	Fun	
		1.4.13.1	Tersedia fungsi unduh data simpanan	Fun	
		1.4.13.2	Tersedia fungsi unduh data angsuran	Fun	
		1.4.13.3	Tersedia fungsi unduh data pinjaman	Fun	
		1.4.13.4	Tersedia fungsi unduh data pinjaman	Fun	

Viewpoint		Requirement			
		1.4.13.4.1	Fungsi unduh hanya dapat digunakan ketika transaksi telah selesai dilakukan	Nfun	
		1.4.13.5	Tersedia fungsi tambah data anggota	Fun	
		1.4.13.6	Tersedia fungsi tambah data angsuran	Fun	
1.5	Bendahara	1.5.1	Memilih opsi simpanan, tarikan, pinjaman dan angsuran	Fun	1.5
		1.5.2	Tersedia fitur Display Data Anggota	Fun	
		1.5.3	Tersedia fitur verifikasi form data tarikan yang telah didapat dari Divisi SP	Fun	
		1.5.4	Tersedia fitur verifikasi form data pinjaman yang telah didapat dari Divisi SP	Fun	
		1.5.5	Tersedia fitur Cari Data Anggota	Fun	
		1.5.6	Tersedia fungsi unduh data	Fun	
		1.5.6.1	Tersedia fungsi unduh data simpanan	Fun	

Viewpoint		Requirement			
		1.5.6.2	Tersedia fungsi unduh data angsuran	Fun	
		1.5.6.3	Tersedia fungsi unduh data pinjaman	Fun	
		1.5.6.4	Tersedia fungsi unduh data pinjaman	Fun	
		1.5.6.5	Fungsi unduh hanya dapat digunakan ketika transaksi telah selesai dilakukan	Nfun	
		1.5.7	Terdapat fungsi untuk mengakses data simpanan wajib	Fun	
		1.5.7.1	Tersedia fungsi cari data simpanan wajib	Fun	
		1.5.7.2	Tersedia fungsi update data simpanan wajib	Fun	
		1.5.7.3	Tersedia fungsi display simpanan wajib		

➤ **Kebutuhan Indirect Viewpoint**

**Tabel 4.6 Indirect Viewpoint**

<b>Viewpoint</b>		<b>Requirement</b>			
<b>Identifier</b>	<b>Label</b>	<b>Description</b>		<b>Type</b>	<b>Souce VP</b>
2.	Lingkungan Koperasi	2.1	Sistem Informasi Koperasi hanya dapat diakses oleh pihak internal koperasi, yaitu anggota, Divisi SP, Ketua Koperasi dan Bendahara Koperasi	Nfun	2
		2.2	Sistem dapat menjalankan fungsi-fungsi yang dibutuhkan pengguna sesuai dengan hak aksesnya	Nfun	2
3.	Peraturan Koperasi	3.1	Sistem informasi koperasi yang dijalankan sepenuhnya dimiliki oleh PT Wonosari Jaya dan hanya dapat dilihat oleh pihak internal koperasi khususnya anggota, Divisi SP, Ketua Koperasi dan Bendahara Koperasi	Nfun	3
3.1	SOP_Koperasi	3.2	Prosedur koperasi yang dimiliki oleh PT Wonosari Jaya	Nfun	3.1

**Keterangan:**

**Fun:** fungsional **Nfun:** nonfungsional

Setelah diketahui apa saja kebutuhan yang dibutuhkan pada Sistem Informasi Koperasi Simpan Pinjam, tahapan selanjutnya adalah menghubungkan antara kebutuhan dengan masing-masing viewpoint serta memberikan deskripsi yang berisi apa saja yang dapat dilakukan oleh sistem pada masing-masing kebutuhan seperti ditunjukkan pada tabel dibawah ini :

**Tabel 4.7 Analisa Kebutuhan beserta deskripsi**

General Type		Service	
No.	Label	Ref.	Description
1.	Login	1.1	Karyawan, Divisi SP, Ketua koperasi dan Bendahara koperasi dapat melakukan fitur <i>login</i> dan <i>logout</i>
2.	Display Data Koperasi	1.1.5	Terdapat fungsi menampilkan data simpanan wajib bagi Anggota
		1.2.1	Tersedia fungsi untuk melihat data simpanan bagi pengguna
		1.2.2	Tersedia fungsi untuk melihat data tarikan bagi pengguna
		1.2.3	Tersedia fungsi untuk melihat data pinjaman bagi pengguna
		1.2.4	Tersedia fungsi untuk melihat data angsuran bagi pengguna
		1.4.2	Tersedia fitur Display Data Anggota bagi Divisi SP
		1.4.11	Tersedia fungsi display data pelunasan pinjaman bagi Divisi SP
		1.5.2	Tersedia fitur Display Data Anggota bagi Bendahara
		1.5.7.3	Tersedia fungsi display simpanan wajib bagi Bendahara
3.	Cari Data	1.3.1	Tersedia fungsi untuk mencari data simpanan bagi pengguna
		1.3.2	Tersedia fungsi untuk mencari data pinjaman bagi pengguna
		1.3.3	Tersedia fungsi untuk mencari data tarikan bagi pengguna

General Type		Service	
		1.3.4	Tersedia fungsi untuk mencari data angsuran bagi pengguna
		1.3.5	Tersedia fitur Cari Data Anggota bagi Ketua
		1.5.5	Tersedia fitur Cari Data Anggota bagi Bendahara
		1.5.7.1	Tersedia fungsi cari data simpanan wajib bagi Bendahara
		1.4.5	Tersedia fitur Cari Data Anggota bagi Divisi SP
		1.4.12	Tersedia fungsi cari data pelunasan pinjaman bagi Divisi SP
4.	Update Data	1.5.7.2	Tersedia fungsi update data simpanan wajib bagi Bendahara
5.	Hak Akses	2.1	Sistem Informasi Koperasi hanya dapat diakses oleh pihak internal koperasi, yaitu karyawan, Divisi SP, Ketua Koperasi dan Bendahara Koperasi
6.	Peraturan Koperasi	3.1	Sistem informasi koperasi yang dijalankan sepenuhnya dimiliki oleh PT Wonosari Jaya dan hanya dapat dilihat oleh pihak internal koperasi khususnya anggota, Divisi SP, Ketua Koperasi dan Bendahara Koperasi

Setelah diketahui apa saja kebutuhan yang dibutuhkan pada Sistem Informasi Koperasi Simpan Pinjam, tahapan selanjutnya adalah menghubungkan antara kebutuhan dengan masing-masing viewpoint seperti ditunjukkan pada tabel dibawah ini :

**Tabel 4.8 Pengelompokan Requirement dengan Viewpoint**

Requirement Number	Viewpoint
1.1	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi, Peraturan Koperasi



Requirement Number	Viewpoint
1.1.5	Anggota dan Bendahara Koperasi
1.2.1	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi,
1.2.2	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi
1.2.3	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi
1.2.4	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi
1.4.2	Divisi SP, Ketua, Bendahara Koperasi
1.4.11	Anggota dan Divisi SP, Peraturan Koperasi
1.5.2	Bendahara Koperasi
1.5.7.3	Anggota, Peraturan Koperasi dan Bendahara Koperasi
1.3.1	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi
1.3.2	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi
1.3.3	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi
1.3.4	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi
1.3.5	Ketua koperasi
1.5.5	Bendahara Koperasi
1.5.7.1	Anggota dan Bendahara Koperasi
1.4.5	Divisi SP
1.4.12	Anggota Peraturan Koperasi dan Divisi SP
1.5.7.2	Anggota Peraturan Koperasi Bendahara Koperasi
2.1	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi, Peraturan Koperasi
3.1	Anggota, Divisi SP, Ketua koperasi dan Bendahara Koperasi, Peraturan Koperasi

Setelah dilakukan list kebutuhan berdasarkan masing-masing viewpoint beserta penjelasannya, tahapan selanjutnya adalah membuat *viewpoint matrix* yang menunjukkan tabel yang berisi nomor kebutuhan dengan hubungan antar relasinya. Yang dimana dilengkapi pula dengan penjelasan apakah kebutuhan tersebut termasuk pada *Indirect Viewpoint* (I) atau *Direct Viewpoint* (D). Hal tersebut telah dijabarkan pada tabel dibawah ini :

	Anggota	Divisi SP	Ketua Koperasi	Bendahara Koperasi	Peraturan Koperasi	Sistem
Anggota		1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.4.2 I 1.4.11 D 1.3.1 D 1.3.2 D 1.3.3 D 1.3.4 D 1.4.5 I 1.4.12 I		1.1.5 D 1.2.1 I 1.2.2 I 1.2.3 I 1.2.4 I 1.5.7.3 D 1.3.1 I 1.3.2 I 1.3.3 I 1.3.4 I 1.5.7.1 I 1.5.7.2 I	1.5.7.3 I 2.1 D 3.1 D	1.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.3.1 D 1.3.2 D 1.3.3 D 1.3.4 D 2.1 D 3.1 D
Divisi SP	1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.4.2 I 1.4.11 D 1.3.1 D 1.3.2 D 1.3.3 D 1.3.4 D 1.4.5 I 1.4.12 I		1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.3.2 D 1.3.3 D 1.3.3 D 1.3.4 D	1.2.1 I 1.2.2 I 1.2.3 I 1.2.4 I 1.5.2 I 1.3.2 I 1.3.3 I 1.3.3 I 1.3.4 I	2.1 D	1.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.4.2 D 1.3.1 D 1.3.2 D 1.3.3 D 1.3.4 D 1.3.5 D 1.4.5 D 1.4.12 I 2.1 D 3.1 D
Ketua Koperasi	1.2.1 I 1.2.2 I 1.2.3 I 1.2.4 I 1.3.1 I 1.3.2 I 1.3.3 I 1.3.4 I	1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.3.2 D 1.3.3 D 1.3.3 D 1.3.4 D		1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D	2.1 D	1.1 D 1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.4.2 D 1.3.1 D 1.3.2 D

	Anggota	Divisi SP	Ketua Koperasi	Bendahara Koperasi	Peraturan Koperasi	Sistem
						1.3.3 D 1.3.4 D 2.1 D 3.1 D
Bendahara Koperasi	1.1.5 D 1.2.1 I 1.2.2 I 1.2.3 I 1.2.4 I 1.5.7.3 D 1.3.1 I 1.3.2 I 1.3.3 I 1.3.4 I 1.5.7.1 I 1.5.7.2 I	1.2.1 I 1.2.2 I 1.2.3 I 1.2.4 I 1.5.2 I 1.3.2 I 1.3.3 I 1.3.3 I 1.3.4 I	1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D		2.1 D 3.1 D	1.1 D 1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.4.2 D 1.5.2 D 1.3.1 D 1.3.2 D 1.3.3 D 1.3.4 D 1.5.5 D 1.5.7.1 D 1.5.7.2 D 2.1 D 3.1 D
Peraturan Koperasi	1.5.7.3 D 2.1 D 3.1 D	2.1 D 3.1 D	2.1 D 3.1 D	2.1 D 3.1 D		
Sistem	1.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.3.1 D 1.3.2 D 1.3.3 D 1.3.4 D 2.1 D 3.1 D	1.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.4.2 D 1.3.1 D 1.3.2 D 1.3.3 D 1.3.4 D 1.3.5 D 1.4.5 D 1.4.12 I 2.1 D 3.1 D	1.1 D 1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.3.1 D 1.3.2 D 1.3.3 D 1.3.4 D 1.3.5 D 1.3.4 D 2.1 D 3.1 D	1.1 D 1.2.1 D 1.2.2 D 1.2.3 D 1.2.4 D 1.5.2 D 1.3.1 D 1.4.2 D 1.3.2 D 1.3.3 D 1.3.4 D 1.5.5 D 1.5.7.1 D 1.5.7.2 D 2.1 D 3.1 D		

Gambar 4.4 Viewpoint Documentation

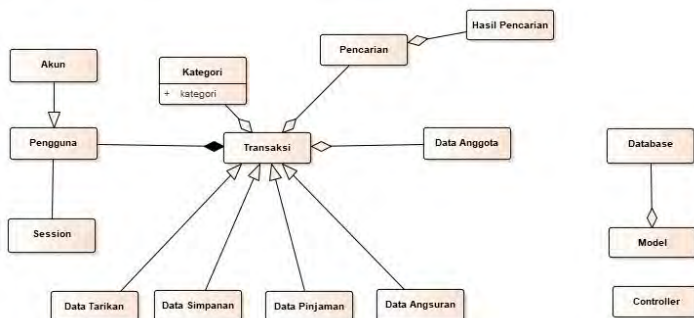
#### 4.2.2 Domain Model

Pada ICONIX process, tahap selanjutnya ialah *Domain Model*. Domain model digunakan untuk menggambarkan obyek-obyek utama yang akan digunakan. Pembuatan domain model dapat diubah seiring dengan pengembangan desain dan aplikasi, sehingga obyek-obyek yang diinisiasikan pada domain model

akan semakin lengkap sesuai dengan kebutuhan pada sistem yang akan dibangun. Dari hasil analisa kebutuhan diatas, didapat beberapa kata benda yang digunakan dalam inisiasi Domain Model yaitu :

1. Transaksi
2. Kategori
3. Data Transaksi
4. Tabel Simpanan
5. Tabel Tarikan
6. Tabel Pinjaman
7. Data Anggota
8. Database
9. Metode Pencarian
10. Halaman Login
11. Pengguna
12. Akun

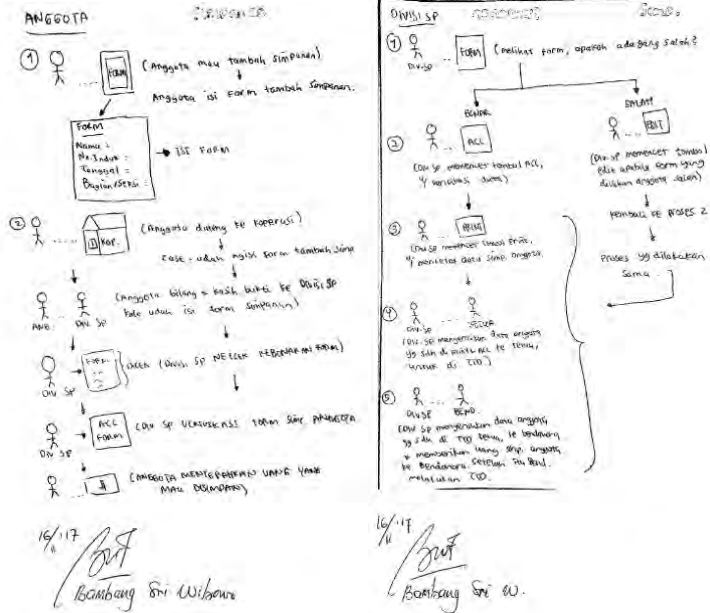
Pada sistem informasi koperasi karyawan PT Wonosari Jaya, penulis menginisialisasi domain model yang diperlihatkan Gambar 4.5 :



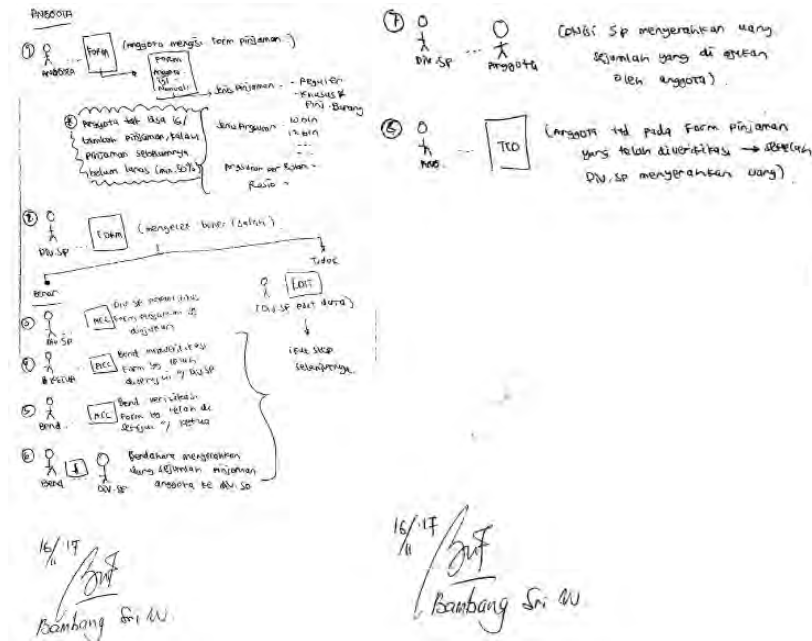
**Gambar 4.5 Domain Model**

### 4.2.3 GUI Storyboard

GUI merupakan sebuah komponen utama dalam sebuah *software*. Storyboard GUI disesuaikan dengan functional requirements didapat pada tahapan wawancara. Dari functional requirements tersebut, peneliti kemudian merancang GUI untuk Sistem Informasi Koperasi. Perancangan GUI Storyboard yang dibuat telah disetujui oleh pegawai koperasi karena GUI Storyboard yang dibuat menceritakan mengenai proses simpan pinjam pada PT Wonosari Jaya. Sebelum merancang GUI Prototype, peneliti membuat GUI Storyboard yang digunakan untuk menceritakan proses dari kebutuhan klien pada sistem informasi koperasi. Gambar dibawah ini merupakan GUI Storyboard Sistem Informasi Koperasi :

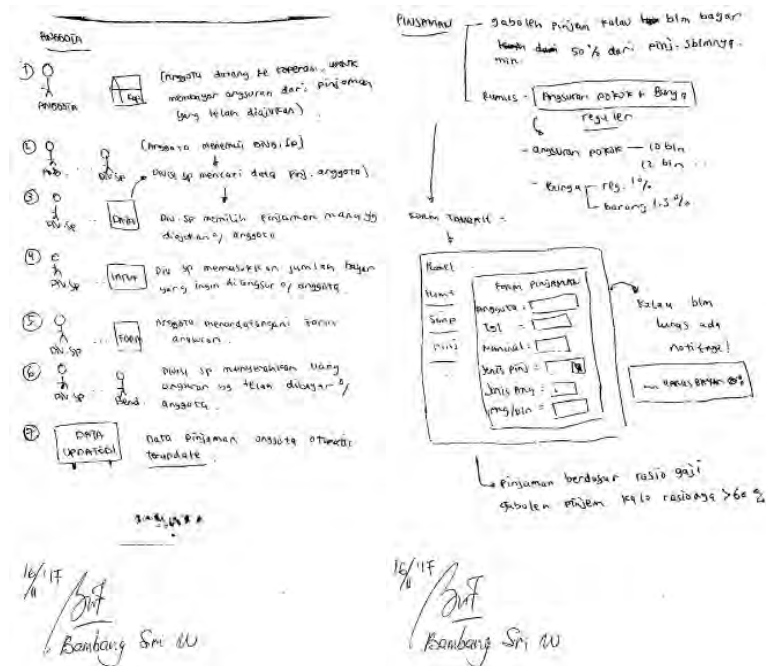


### Gambar 4.7 GUI Storyboard Tarikan

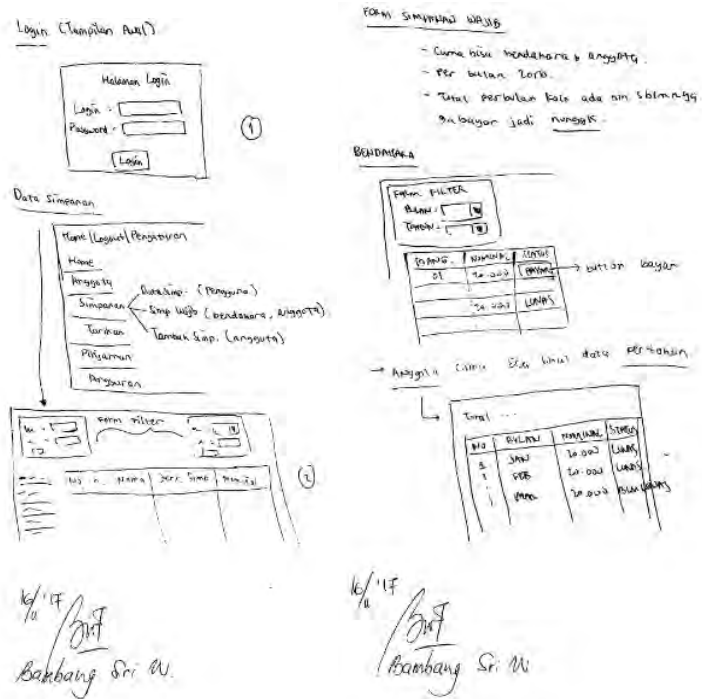


Gambar 4.8 GUI Storyboard Pinjaman





Gambar 4.9 GUI Storyboard Angsuran



Gambar 4.10 GUI Storyboard Tampilan

Setelah membuat GUI Storyboard, langkah selanjutnya adalah membuat GUI Prototype. GUI Prototype yang dibuat pada penelitian ini menggunakan tools “Mockplus”. Dibawah ini merupakan beberapa contoh dari GUI Prototype pada sistem informasi koperasi :

#### **a. Halaman Login**

Form Login merupakan form awal yang menentukan apakah pengguna dapat mengakses aplikasi koperasi. Pada form login ini, pengguna yang salah memasukkan username atau password akan kembali ke halaman login. Terdapat empat pengguna aplikasi koperasi, yaitu Anggota, Divisi SP, Ketua, dan Bendahara Koperasi.



The image shows a mockup of a login page. At the top, the title "Halaman Login" is displayed in a green font. Below the title, there are two input fields. The first field is labeled "Login" followed by a colon, and the second field is labeled "Password" followed by a colon. Both labels are in a green font. At the bottom of the form, there is a button labeled "Masuk" in a green font.

**Gambar 4.11 Halaman Login**

#### **b. Halaman Data Simpanan Wajib**

Halaman Data Simpanan Wajib dapat diakses oleh Anggota dan Bendahara. Didalamnya terdapat data pembayaran simpanan wajib per tahun anggota.

Data Simpanan Wajib				
Menu   Logout   Selamat Datang : Anggota   Hak Akses : Anggota				
Home	Total yang perlu dibayar bulan ini adalah Rp 60.000,00			
Simpanan				
Data Simpanan				
Data Simpanan Wajib				
Tambah Simpanan				
Tarikan				
Pinjaman				
Angsuran				
NO	BULAN	NOMINAL	STATUS	
1.	JANUARI	Rp 20.000	LUNAS	
2.	FEBRUARI	Rp 20.000	LUNAS	
3.	MARET	Rp 20.000	LUNAS	
4.	APRIL	Rp 20.000	LUNAS	
5.	MAY	Rp 20.000	LUNAS	
6.	JUNI	Rp 20.000	LUNAS	
7.	JULI	Rp 20.000	LUNAS	

Gambar 4.12 GUI Data Simpanan Wajib

### c. Halaman Tambah Simpanan

Halaman Tambah Simpanan hanya dapat diakses oleh Anggota. Anggota wajib mengisi semua data yang dibutuhkan, ketika semua data telah terisi maka anggota dapat menekan tombol simpan. Jika data yang dimasukkan tidak lengkap dan tetap meng-klik tombol simpan maka akan muncul pesan untuk melengkapi data.

Form Simpanan	
Menu   Logout   Selamat Datang : Anggota   Hak Akses : Anggota	
Home	
Simpanan	
Data Simpanan	
Data Simpanan Wajib	
Tambah Simpanan	
Tarikan	
Pinjaman	
Angsuran	

**Form Simpanan**

Anggota : AKN-0001

Tanggal : 03 / 10 / 2014

Nominal :

Jenis Simpanan : Simpanan Sukarela  
Simpanan Sukarela  
Simpanan Pokok

**SIMPAN** **BATAL**

Gambar 4.13 GUI Form Tambah Simpanan

#### d. Tambah Pengguna

Halaman Tambah Pengguna hanya dapat diakses oleh Divisi SP untuk membuat akun anggota koperasi. Divisi SP wajib mengisi semua data yang dibutuhkan, ketika semua data telah terisi maka Divisi SP dapat menekan tombol simpan. Jika data yang dimasukkan tidak lengkap dan tetap menekan tombol simpan maka akan muncul pesan untuk melengkapi data.

The screenshot shows a web application interface for adding a new user. The title bar at the top reads 'Form Anggota'. Below it, a navigation bar contains links for 'Menu', 'Logout', 'Selamat Datang : Divisi SP', and 'Hak Akses : Divisi SP'. A sidebar menu on the left lists various options: 'Home', 'Anggota' (highlighted), 'Data Anggota', 'Tambah Anggota', 'Simpanan', 'Tarikan', 'Pinjaman', and 'Angsuran'. The main content area is titled 'Form Tambah Pengguna' and contains several input fields: 'ID Anggota', 'Nama Anggota', 'Username', 'Password', 'Tanggal Masuk' (with a date picker showing 'hh/bb/yyyy'), 'Tanggal Lahir' (with a date picker showing 'hh/bb/yyyy'), 'Alamat', 'No. Telp', 'Gaji', 'Jabatan', 'Seksi/Unit', and 'Role' (a dropdown menu currently set to 'Anggota'). At the bottom of the form, there are two green buttons labeled 'SIMPAN' and 'BATAL'.

**Gambar 4.14 Form Tambah Pengguna**

### 4.1.4 Use Case

#### 4.1.4.1 Deskripsi Use Case

Setelah melakukan analisa maka didapatkan kebutuhan fungsional yang dikelompokkan berdasarkan pengguna. Kebutuhan fungsional tersebut antara lain:

1. Pengguna dapat melakukan fitur *login* dan *logout*
2. Pengguna dapat menggunakan fitur display data untuk melihat data-data transaksi yang telah tersimpan
3. Pengguna dapat menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi
4. Divisi SP dapat menggunakan fitur edit setelah anggota melakukan transaksi. Divisi SP dapat melakukan pengubahan data dengan menggunakan fitur edit disebelah kanan tabel transaksi. Divisi SP hanya dapat mengedit nama data anggota, data simpanan, data tarikan, dan data pinjaman.
5. Divisi SP memiliki kewenangan untuk memverifikasi data simpanan setelah anggota melakukan tambah simpanan
6. Data pelunasan pinjaman hanya dapat diakses oleh Divisi SP. Divisi SP memiliki kewenangan untuk melihat data pelunasan pinjaman.
7. Didalam data pelunasan peminjaman, terdapat fitur cari data yang digunakan untuk memudahkan Divisi SP dalam mencari data transaksi.
8. Didalam data pelunasan peminjaman, terdapat fitur cari data yang digunakan untuk memudahkan Divisi SP dalam mencari data transaksi.
9. Divisi SP memiliki kewenangan untuk menggunakan fitur unduh data. Fitur tersebut digunakan oleh Divisi SP untuk diteruskan ke tahap selanjutnya. Data yang telah diunduh selanjutnya dikirimkan ke Ketua untuk di tanda tangani.
10. Divisi SP, Bendahara dan Ketua dapat menggunakan fitur display data untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi. Selanjutnya, Divisi SP, Ketuadan Bendahara dapat melakukan verifikasi data apabila data yang diajukan oleh anggota sudah benar.
11. Bendahara memiliki tugas untuk mengatur data simpanan wajib anggota. Oleh karena itu, Bendahara memiliki satu

fitur yaitu “Cari Data” yang memudahkan Bendahara dalam mencari data simpanan wajib anggota.

12. Bendahara memiliki tugas untuk mengatur data simpanan wajib anggota. Hanya bendahara yang memiliki kewenangan untuk mengupdate data simpanan wajib anggota. Data simpanan wajib anggota bersifat bulanan.
13. Anggota memiliki peran penting dalam system informasi koperasi. Oleh karena itu, anggota memiliki fitur tambah data yang dapat digunakan untuk menambah data simpanan, data tarikan dan data pinjaman. Fitur tambah data hanya dapat digunakan ketika anggota ingin melakukan transaksi di koperasi.
14. Divisi SP memiliki peran utama dalam sistem informasi koperasi. Karena, sebagian besar data dikelola oleh Divisi SP. Oleh karena itu, Divisi SP memiliki kewenangan untuk menambah data anggota yang didalamnya terdiri dari Nama Anggota, username dan password (digunakan anggota untuk melakukan transaksi). Fitur tambah data yang digunakan oleh Divisi SP adalah pada fitur tambah data anggota dan tambah data angsuran.

#### 4.1.4.2 Aktor-aktor yang terlibat

Aktor adalah kesatuan eksternal yang berinteraksi langsung dengan usecase. Aktor-aktor yang terlibat dengan perangkat lunak koperasi karyawan ini diantaranya adalah :

1. **Pengguna**, merupakan pengguna dari sistem secara umum. Pengguna disini dibagi menjadi 4 bagian, yaitu Karyawan,, Ketua Koperasi dan Bendahara Koperasi. Empat aktor turunan ini memiliki fungsi yang sama
2. **Anggota**, merupakan aktor turunan dari pengguna yang dapat melakukan Login, memilih opsi simpanan, pinjaman, tarikan dan angsuran uang, melakukan pendaftaran simpanan, pinjaman, tarikan, dan angsuran uang, menerima bukti telah melakukan proses transaksi dan melakukan Logout

3. **Divisi SP**, merupakan aktor turunan dari pengguna yang dapat melakukan *Login*, mengubah password pengguna, melihat daftar karyawan, memvalidasi form yang diajukan oleh karyawan, mengelola data karyawan, menggunakan *search engine* untuk mencari data karyawan, memberikan rekap bukti penyerahan uang (pinjaman) , mengunduh data transaksi dan menyerahkan kepada anggota, ketua dan bendahara
4. **Ketua Koperasi**, merupakan aktor turunan dari pengguna yang dapat melakukan *Login*, melihat daftar karyawan, memvalidasi form yang diajukan oleh Divisi SP, menggunakan *search engine* untuk mencari data karyawan dan melakukan *Logout*
5. **Bendahara Koperasi**, merupakan aktor turunan dari pengguna yang dapat melakukan *Login*, memvalidasi form yang diajukan oleh ketua koperasi, memegang wewenang dalam proses simpanan wajib, menggunakan *search engine* untuk mencari data karyawan dan melakukan *Logout*.

Dari ilustrasi di atas dapat digambarkan generalisasi aktor yang terlibat dalam sistem informasi koperasi karyawan PT Wonosari Jaya yang dapat dilihat pada :



**Gambar 4.15 Generalisasi Aktor**

#### 4.1.4.3 Memodelkan Use Case

Tahap ini merupakan tahap dalam pembuatan use case yang ditentukan berdasarkan kebutuhan fungsional dari sebuah sistem informasi koperasi karyawan PT Wonosari Jaya yang



akan dibuat. Berikut ini merupakan hasil turunan dari kebutuhan fungsional yang akan menjadi use case :

**Tabel 4.9 Turunan Kebutuhan Fungsional menjadi Use Case**

<b>Kebutuhan Fungsional</b>	<b>Use Case</b>
KF01. Fitur Autentifikasi	<b>UC1.1</b> Login <b>UC1.2</b> Logout
KF02. Terdapat fitur display data dan cari data yang dilakukan oleh semua pengguna	<b>UC2.1</b> Display Data Simpanan <b>UC2.2</b> Display Data Tarikan <b>UC2.3</b> Display Data Pinjaman <b>UC2.4</b> Display Data Angsuran <b>UC2.5</b> Cari Data Simpanan <b>UC2.6</b> Cari Data Pinjaman <b>UC2.7</b> Cari Data Tarikan <b>UC2.8</b> Cari Data Angsuran
KF03 Terdapat fitur pengubahan data yang dilakukan oleh Divisi SP	<b>UC3.1</b> Edit data Anggota <b>UC3.2</b> Edit Data Simpanan <b>UC3.3</b> Edit Data Tarikan <b>UC3.2</b> Edit Data Pinjaman
KF04 Terdapat fitur verifikasi yang hanya dapat dilakukan oleh Divisi SP	<b>UC4.1</b> Verifikasi Data Simpanan
KF05 Terdapat fitur display data dan cari data yang hanya dapat dilakukan oleh Divisi SP	<b>UC5.1</b> Display Data Pelunasan Pinjaman <b>UC5.2</b> Cari Data Pelunasan Pinjaman
KF06 Terdapat fitur unduh yang hanya dapat digunakan oleh Bendahara dan Divisi SP	<b>UC6.1</b> Unduh Data Simpanan <b>UC 6.2</b> Unduh Data Angsuran <b>UC 6.3</b> Unduh Data Pinjaman <b>UC 6.4</b> Unduh Data Tarikan
KF07 Terdapat fitur display data, cari data dan	<b>UC7.1</b> Display Data Anggota <b>UC7.2</b> Verifikasi Data Tarikan <b>UC7.3</b> Verifikasi Data Pinjaman

<b>Kebutuhan Fungsional</b>	<b>Use Case</b>
verifikasi data yang hanya digunakan oleh Divisi SP, Ketua dan Bendahara	<b>UC 7.4</b> Cari Data Anggota
KF08 Terdapat fitur display dan cari data yang hanya dapat digunakan oleh Bendahara dan Anggota	<b>UC8.1</b> Display Data Simpanan Wajib
KF09 Terdapat fitur update data yang hanya dapat digunakan oleh Bendahara	<b>UC9.1</b> Cari Data Simpanan Wajib <b>UC9.2</b> Update Data Simpanan Wajib
KF10 Terdapat fitur tambah data yang hanya dapat dilakukan oleh anggota	<b>UC10.1</b> Tambah Data Simpanan <b>UC10.2</b> Tambah Data Tarikan <b>UC10.3</b> Tambah Data Pinjaman
KF11 Terdapat fitur tambah data yang hanya dapat dilakukan oleh divisi SP	<b>UC11.1</b> Tambah Data Anggota <b>UC11.2</b> Tambah Daftar Tagihan Angsuran
KF12 Terdapat fitur pengaturan akun yang dapat diakses oleh semua pengguna	<b>UC12.1</b> Edit Akun

### **Detail Use Case**

Untuk memudahkan dalam proses pembuatan aplikasi, dibuatlah poin-poin yang diharapkan dari masing-masing use

case. Berikut ini merupakan detail dari masing-masing use case berdasarkan penjelasannya :

**Tabel 4.10 Detail UseCase**

<b>Use Case</b>	<b>Detail Use Case</b>
<b>UC1.1</b> Login	Error Message untuk proses login
<b>UC1.2</b> Logout	Kembali ke Halaman Login.
<b>UC2.1</b> Display Data Simpanan	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel.</li> <li>• Terdapat fitur cari</li> </ul>
<b>UC2.2</b> Display Data Tarikan	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> </ul>
<b>UC2.3</b> Display Data Pinjaman	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> </ul>
<b>UC2.4</b> Display Data Angsuran	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> </ul>
<b>UC2.5</b> Cari Data Simpanan	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> <li>• Dapat difilter sesuai kata kunci yang dimasukkan</li> </ul>
<b>UC2.6</b> Cari Data Pinjaman	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> <li>• Dapat difilter sesuai kata kunci yang dimasukkan</li> </ul>
<b>UC2.7</b> Cari Data Tarikan	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> <li>• Dapat difilter sesuai kata kunci yang dimasukkan</li> </ul>

Use Case	Detail Use Case
<b>UC2.8</b> Cari Data Angsuran	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> <li>• Dapat difilter sesuai kata kunci yang dimasukkan</li> </ul>
<b>UC3.1</b> Edit data Anggota	<ul style="list-style-type: none"> <li>• Data yang diedit sesuai dengan data yang dipilih pada baris tabel data user</li> <li>• Form edit meliputi ID Anggota, Tanggal Masuk, Tanggal Lahir, No Pegawai, Alamat, No Telp, Gaji, Jabatan, Seksi/Unit</li> <li>• Form menampilkan data sebelum diedit</li> <li>• Terdapat notifikasi sukses dan error</li> </ul>
<b>UC3.2</b> Edit Data Simpanan	<ul style="list-style-type: none"> <li>• Data yang diedit sesuai dengan data yang dipilih pada baris tabel data user</li> <li>• Form edit meliputi ID Anggota, Tanggal, Nama Anggota, Nominal, Jenis Simpanan</li> <li>• Form menampilkan data sebelum diedit</li> <li>• Terdapat notifikasi sukses dan error</li> </ul>

Use Case	Detail Use Case
<b>UC3.3</b> Edit Data Tarikan	<ul style="list-style-type: none"> <li>• Data yang diedit sesuai dengan data yang dipilih pada baris tabel data user</li> <li>• Form edit meliputi ID Tarikan, ID Anggota, Tanggal, Nama Anggota, Nominal</li> <li>• Form menampilkan data sebelum diedit</li> <li>• Terdapat notifikasi sukses dan error</li> </ul>
<b>UC3.2</b> Edit Data Pinjaman	<ul style="list-style-type: none"> <li>• Data yang diedit sesuai dengan data yang dipilih pada baris tabel data user</li> <li>• Form edit meliputi ID Pinjaman, Jenis Pinjaman, Tanggal, Jenis Angsuran, Nominal</li> <li>• Form menampilkan data sebelum diedit</li> <li>• Terdapat notifikasi sukses dan error</li> <li>• Terdapat fitur reset data</li> </ul>
<b>UC4.1</b> Verifikasi Data Simpanan	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur Verifikasi</li> </ul>
<b>UC5.1</b> Display Data Pelunasan Pinjaman	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk table</li> <li>• Terdapat fitur cari</li> </ul>
<b>UC5.2</b> Cari Data Pelunasan Pinjaman	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> <li>• Dapat difilter sesuai kata kunci yang dimasukkan</li> </ul>
<b>UC6.1</b> Unduh Data Simpanan <b>UC 6.2</b> Unduh Data Angsuran <b>UC 6.3</b> Unduh Data Pinjaman <b>UC 6.4</b> Unduh Data Tarikan	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur unduh</li> </ul>

Use Case	Detail Use Case
<b>UC7.1</b> Display Data Anggota	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk table</li> <li>• Terdapat fitur cari</li> </ul>
<b>UC7.2</b> Verifikasi Data Tarikan	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur Verifikasi</li> </ul>
<b>UC7.3</b> Verifikasi Data Pinjaman	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur Verifikasi</li> </ul>
<b>UC 7.4</b> Cari Data Anggota	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> <li>• Dapat difilter sesuai kata kunci yang dimasukkan</li> </ul>
<b>UC8.1</b> Display Data Simpanan Wajib	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk table</li> <li>• Terdapat fitur cari</li> </ul>
<b>UC9.1</b> Cari Data Simpanan Wajib	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur cari</li> <li>• Dapat difilter sesuai kata kunci yang dimasukkan</li> </ul>
<b>UC9.2</b> Update Data Simpanan Wajib	<ul style="list-style-type: none"> <li>• Data ditampilkan dalam bentuk tabel</li> <li>• Terdapat fitur “Bayar” dalam bentuk button</li> </ul>

Use Case	Detail Use Case
<b>UC10.1</b> Tambah Data Simpanan	<ul style="list-style-type: none"> <li>• Terdapat form yang berisi ID Anggota, Tanggal, Nominal, Jenis Simpanan</li> <li>• Terdapat fitur pilih jenis simpanan</li> <li>• Terdapat tombol simpan, dan terdapat notifikasi apabila anggota tidak mengisi semua data</li> <li>• Terdapat tombol batal</li> </ul>
<b>UC10.2</b> Tambah Data Tarikan	<ul style="list-style-type: none"> <li>• Terdapat form yang berisi ID Anggota, Tanggal, Nominal, dan Sisa Saldo</li> <li>• Terdapat tombol simpan, dan terdapat notifikasi apabila anggota tidak mengisi semua data</li> <li>• Terdapat tombol batal</li> </ul>
<b>UC10.3</b> Tambah Data Pinjaman	<ul style="list-style-type: none"> <li>• Terdapat form yang berisi ID Anggota, Tanggal, Nominal, Jenis Pinjaman, Jenis Angsuran, Angsuran per bulan dan Rasio</li> <li>• Terdapat tombol simpan, dan terdapat notifikasi apabila anggota tidak mengisi semua data</li> <li>• Terdapat tombol batal</li> </ul>

Use Case	Detail Use Case
<b>UC11.1</b> Tambah Data Anggota	<ul style="list-style-type: none"> <li>• Terdapat form yang berisi ID Anggota, Tanggal Masuk, Tanggal Lahir, No Pegawai, Alamat, No Telp, Gaji, Jabatan, Seksi/Unit</li> <li>• Terdapat tombol simpan, dan terdapat notifikasi apabila anggota tidak mengisi semua data</li> <li>• Terdapat tombol batal</li> </ul>
<b>UC11.2</b> Tambah Daftar Tagihan Angsuran	<ul style="list-style-type: none"> <li>• Terdapat form yang berisi ID Pinjaman Belum Lunas, Tanggal Angsuran, Jumlah Bayar</li> <li>• Terdapat tombol simpan, dan terdapat notifikasi apabila anggota tidak mengisi semua data</li> <li>• Terdapat tombol batal</li> </ul>
<b>UC12.1</b> Edit Akun	<ul style="list-style-type: none"> <li>• Terdapat form ubah nama</li> <li>• Terdapat form berisi password baru dan ulangi password baru</li> <li>• Terdapat tombol simpan, dan terdapat notifikasi apabila anggota tidak mengisi semua data</li> <li>• Terdapat tombol batal</li> </ul>

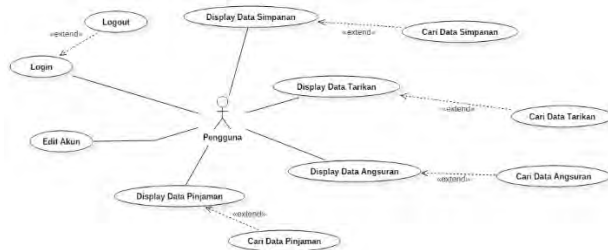




### ***Use Case Diagram Berdasarkan Aktor***

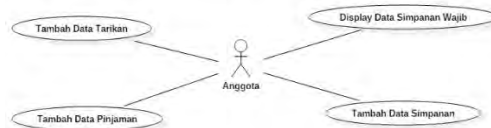
Gambar dibawah ini merupakan hubungan antara masing-masing usecase dengan masing-masing aktor :

#### **1. Use Case diagram Pengguna**



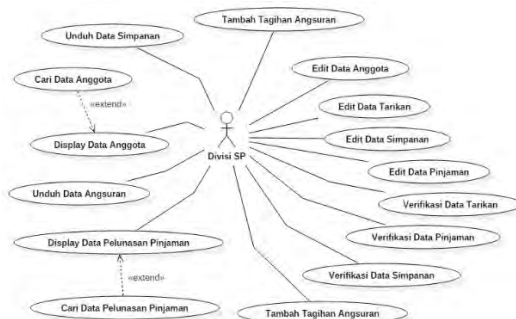
**Gambar 4.17 Use Case Diagram Pengguna**

#### **2. Use Case diagram Anggota**



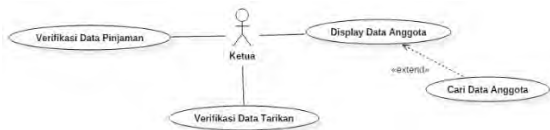
**Gambar 4.18 Use case diagram anggota**

#### **3. Use case diagram Divisi SP**



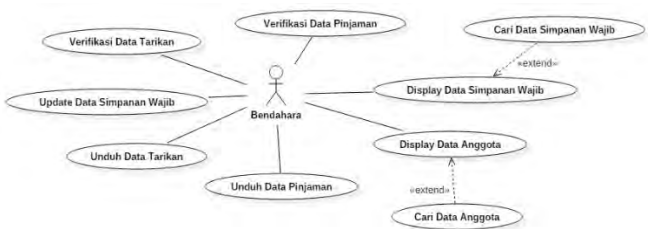
**Gambar 4.19 Use case diagram Divisi SP**

4. Use case diagram Ketua



Gambar 4.20 Use case diagram Ketua

5. Use case diagram Bendahara



Gambar 4.21 Use case diagram Bendahara

4.1.4.5 Use Case Description

Deskripsi sebuah use case (Use Case Description) berisikan tentang perilaku yang ada pada use case. Berikut ini merupakan salah satu contoh bentuk deskripsi use case yang terdapat pada aplikasi Koperasi Simpan Pinjam PT Wonosari Jaya :

- Pengguna

Tabel 4.11 Use Case Description Login

+	UC.01.01 Login
Ringkasan :	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melakukan <i>login</i> sehingga dapat menggunakan fasilitas-fasilitas yang sesuai dengan hak akses masing-masing.
Direct Aktor :	Anggota, Divisi SP, Ketua, Bendahara

+	UC.01.01 Login			
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali			
<b>Pre Condition :</b>	-			
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem	
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login	
	1.1	Pengguna memasukkan password dan username		
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal	
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika username salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> <li>2. Jika password salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> <li>3. Jika username dan password tidak diisi, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> <li>4. Jika username tidak ditemukan, maka sistem akan memberikan pemberitahuan</li> </ol>			

+	UC.01.01 Login
	“Username dan Password yang anda masukkan tidak sesuai!”

- Anggota

**Tabel 4.12 Use Case Description "Tambah Data Simpanan"**

+	UC.10.01 Tambah Data Simpanan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat melakukan pendaftaran transaksi dari sistem informasi koperasi simpan pinjam		
<b>Direct Aktor :</b>	Anggota		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1	Aktor mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Aktor memasukkan password dan username	
	1.2	Aktor menekan button “Masuk”	Sistem menampilkan halaman awal
	3	Aktor memilih fitur simpanan	Sistem menampilkan halaman simpanan

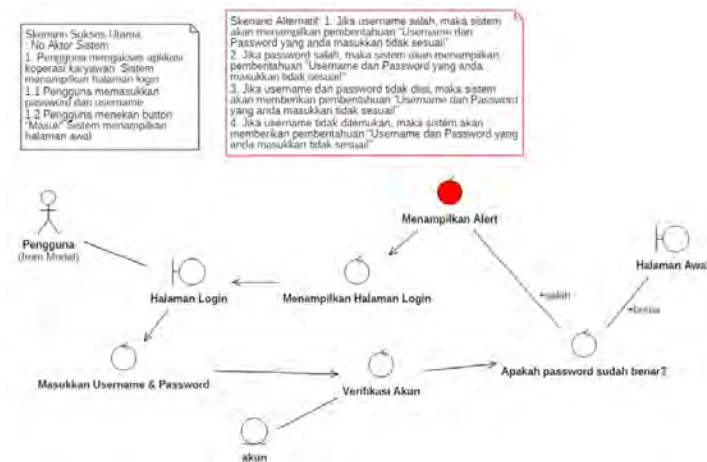
+	UC.10.01 Tambah Data Simpanan		
	3.1	Aktor memilih fitur “tambah simpanan”	Sistem menampilkan halaman tambah simpanan
	3.1.1	Aktor mengisi data tambah simpanan	
	3.1.2	Aktor menekan tombol simpan	Sistem melakukan pengecekan apakah semua field sudah diisi dan jumlah transaksi yang dimasukan merupakan bilangan asli positif
	3.1.3	Aktor melihat data hasil registrasi	Sistem menampilkan halaman data simpanan yang berisi detail anggota serta status apakah data simpanan telah disetujui
<b>Skenario Alternatif:</b>		<ol style="list-style-type: none"> <li>1. Sistem akan menampilkan halaman simpanan kembali apabila anggota memencet tombol batal</li> <li>2. Sistem akan menampilkan halaman yang berisi notifikasi bahwa registrasi gagal dilakukan karena pengisian data tidak lengkap</li> <li>3. Sistem akan menampilkan pesan “tidak boleh kosong” disebelah kanan <i>field</i> jika anggota belum melengkapi form</li> </ol>	

Untuk Use Case description lebih lengkap dapat dilihat pada lampiran A.

## 4.2 Analysis and Preliminary Design

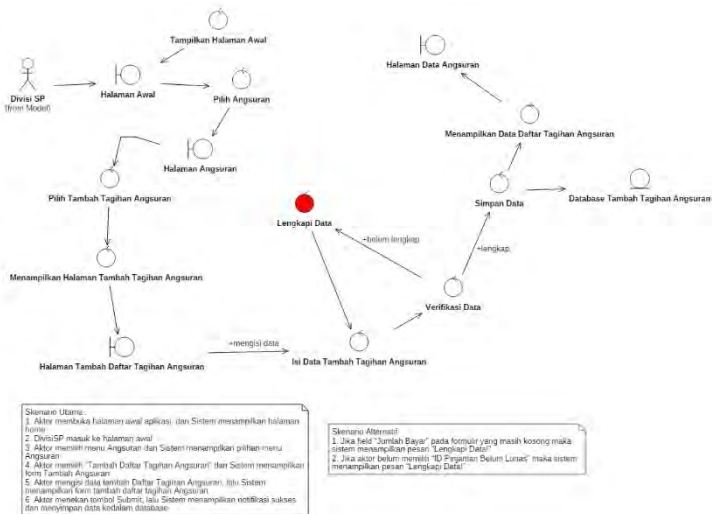
Proses berikutnya adalah melakukan robustness analisis terhadap object yang ada pada use case tersebut. Objek tersebut adalah boundary, entity, dan controller. *Input* dari pembuatan Robustness Diagram adalah hasil dari *Use Case Description*. Dari robustness analisis ini nantinya akan menghasilkan robust diagram pada sistem informasi koperasi PT Wonosari Jaya :

### 1. Robustness Diagram **Login** berdasarkan *Use Case Diagram Login*



Gambar 4.22 Robustness Diagram Login

## 2. Robustness Diagram Tambah Tagihan Angsuran berdasarkan Use Case Diagram Tambah Tagihan Angsuran



**Gambar 4.23 Robustness Tambah Tagihan Angsuran**

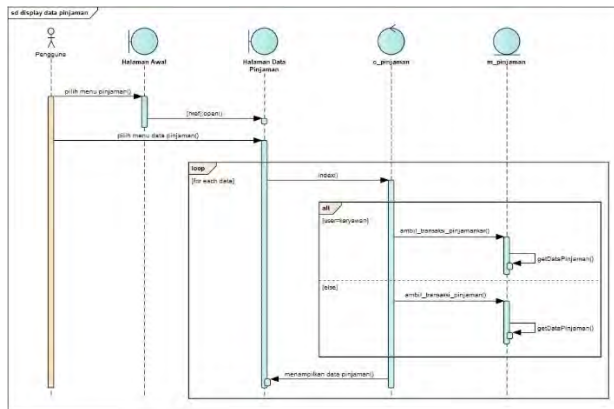
Untuk Robustness Diagram lebih lengkap dapat dilihat pada Lampiran B.

## 4.3 Sequence Diagram

*Detailed design (perancangan secara rinci)* dari pembuatan suatu software ditunjukkan melalui pemodelan *sequence diagram*. Berikut ini merupakan *sequence diagram* pada sistem informasi koperasi simpan pinjam PT Wonosari Jaya berdasarkan *robustness diagram* :

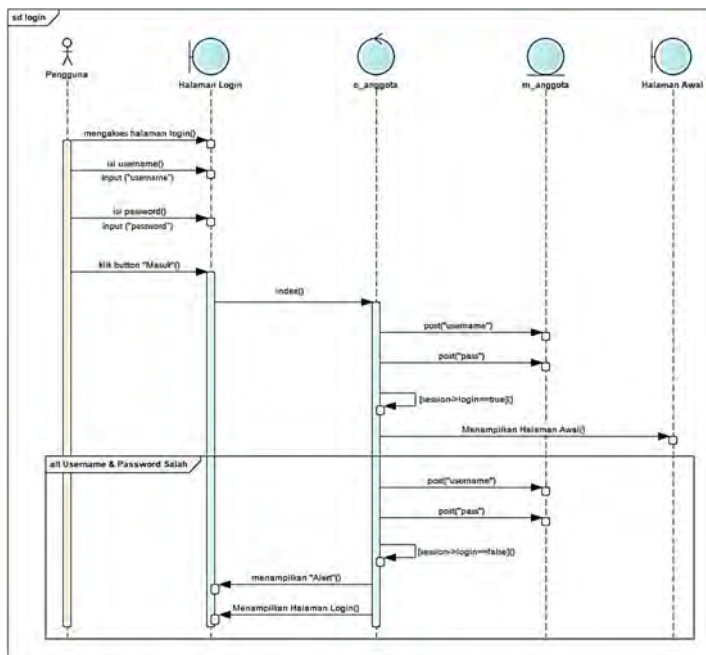


## 1. Sequence Diagram Display Data Pinjaman



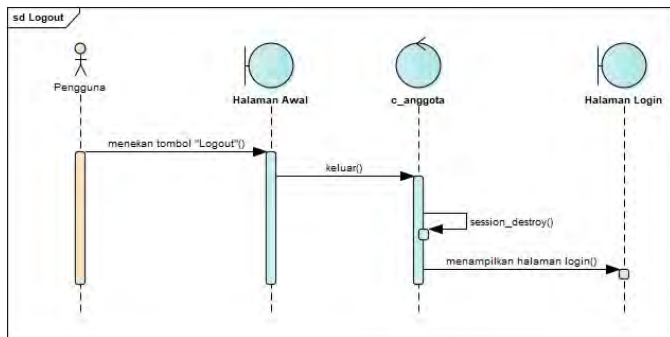
**Gambar 4.24. Sequence Diagram Display Data Pinjaman**

## 2. Sequence Diagram Login



**Gambar 4.25 Sequence Diagram Login**

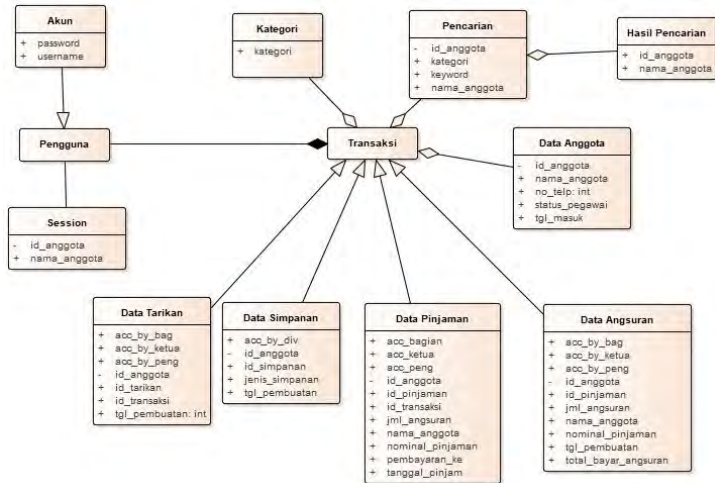
### 3. Sequence Diagram Logout



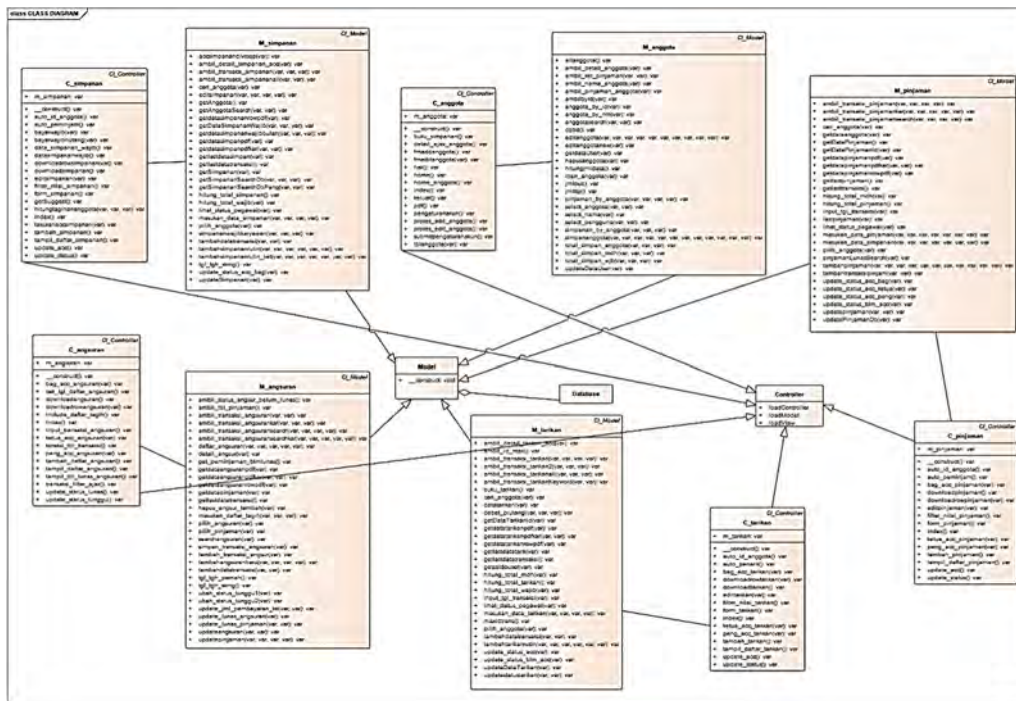
Gambar 4.26 Sequence Diagram Logout

## 4.4 Class Diagram

Dari apa yang telah dilakukan pada *sequence diagram*, nantinya mungkin terdapat perubahan-perubahan yang terdapat pada domain model sehingga perlu adanya *update domain model*. Semakin detail gambaran proses yang terdapat pada *robustness analysis* dan *sequence diagram*, maka akan semakin memudahkan dalam pembuatan *update domain model*. *Update domain model* yang telah dibuat, selanjutnya akan dibuat menjadi *class diagram*, yang dimana didalamnya terdapat atribut-atribut dari objek yang ada dan juga operasi yang dapat dilakukan pada objek tersebut. Dibawah ini merupakan *Class Diagram* dari sistem informasi koperasi simpan pinjam PT Wonosari Jaya yang telah di *update* dari domain model yang telah dibuat sebelumnya :



Gambar 4.27 Class Diagram (1)



*“Halaman ini sengaja dikosongkan”*

## IMPLEMENTASI

Pada bagian ini akan dijelaskan mengenai implementasi pengembangan aplikasi. Adapun subbab dari bab impelmentasi yaitu: tahapan implemenasi dan implementasi.

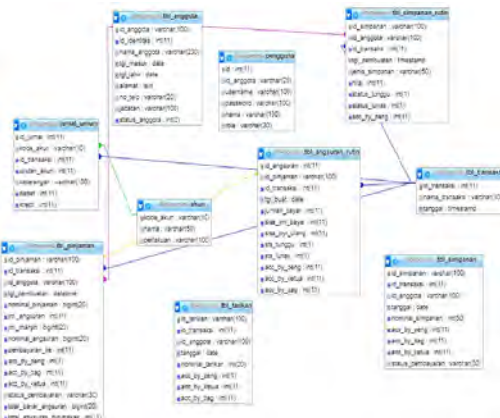
## 5.1 Implementasi

Adapun tahapan yang dilakukan oleh peneliti dalam melakukan implementasi aplikasi dengan studi kasus Koperasi Karyawan PT Wonosari Jaya adalah sebagai berikut :

### 5.1.1 Pembuatan database

Database dibuat berdasarkan desain model data fisik . Desain data model fisik yang sudah dibuat akan langsung degenerate kedalam bentuk database spesifik. Pada pembuatan aplikasi koperasi karyawan ini, peneliti menggunakan database yang dibuat dalam bentuk database MySQL.

Gambar 5.1 merupakan desain database dari aplikasi koperasi simpan pinjam PT Wonosari Jaya :



**Gambar 5.1 Database Sistem Informasi Koperasi Karyawan PT Wonosari Jaya**

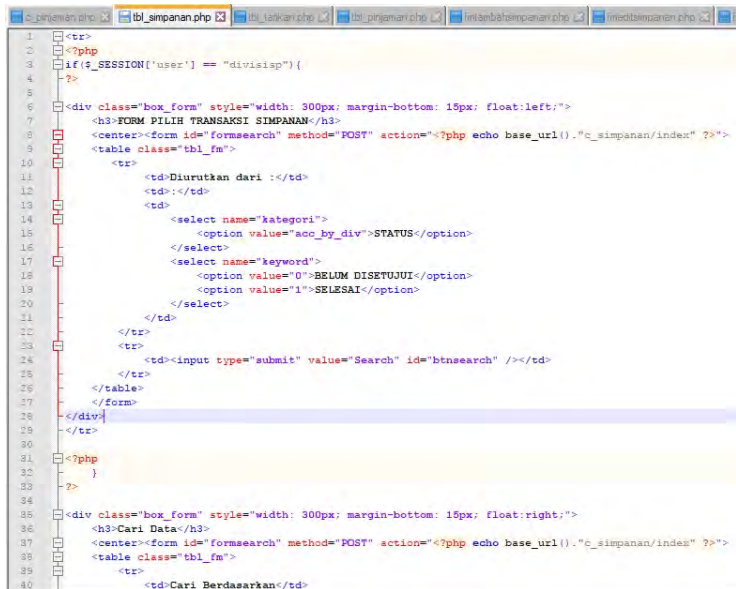
### 5.2.2 Pengkodean/Implementasi

Pengkodean/implementasi dilakukan setelah database selesai dibuat. Pada pengkodean ini, peneliti menggunakan bahasa pemrograman PHP dengan framework CodeIgniter. Peneliti dalam melakukan pengkodean ini menggunakan konsep MVC yang kepanjangannya *Model View Controller*. Kode-kode program yang berhubungan langsung dengan database, seperti menjalankan query akan masuk pada folder/ bagian dari model. Sedangkan kode program yang berhubungan dengan tampilan seperti kode HTML akan masuk pada bagian *View*. Controller berisikan kode-kode program yang menghubungkan *View* dengan *Model*. Berikut ini adalah salah satu contoh file kode yang ada pada aplikasi koperasi simpan pinjam dibawah ini :

a. Sampel kode aplikasi

Sampel kode aplikasi digunakan sebagai acuan dalam pembuatan Sequence Diagram. Didalam Sequence Diagram, terdapat *Boundary*, *Controller* dan *Model*. Pada *Boundary* Sequence Diagram, digunakan view pada pengkodean yang menghasilkan input sebagai GUI. Selain itu, *Controller* dan *Entity* pada Sequence Diagram mengacu pada *Controller* pada pengkodean yaitu dengan nama folder *c\_simpanan* dan *Entity* pada pengkodean yaitu dengan nama folder *m\_simpanan*

Gambar 5.2 – 5.4 merupakan sample dari kode pada Sequence Diagram Display Data Simpanan.

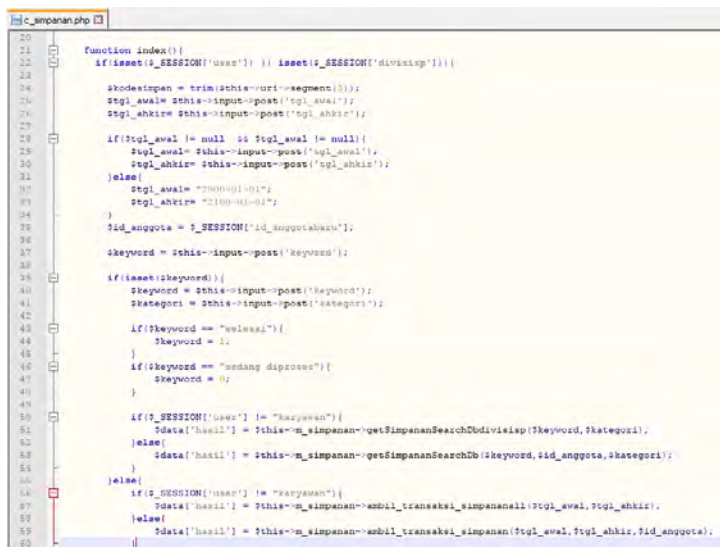


```

1  <tr>
2  <?php
3  if(!$_SESSION['user'] == "divisiap"){
4  ?>
5
6  <div class="box_form" style="width: 300px; margin-bottom: 15px; float:left;">
7  <h3>FORM PILIH TRANSAKSI SIMPANAN</h3>
8  <center><form id="formsearch" method="POST" action="<?php echo base_url(). "c_simpanan/index" ?>">
9  <table class="tbl_fm">
10 <tr>
11 <td>Biurutkan dari :</td>
12 <td></td>
13 <td>
14 <select name="kategori">
15 <option value="acc_by_div">STATUS</option>
16 </select>
17 <select name="keyword">
18 <option value="0">BELUM DISETUJUI</option>
19 <option value="1">SELESAI</option>
20 </select>
21 </td>
22 </tr>
23 <tr>
24 <td><input type="submit" value="Search" id="btnsearch" /></td>
25 </tr>
26 </table>
27 </form>
28 </div>
29 </tr>
30
31 <?php
32 }
33 ?>
34
35 <div class="box_form" style="width: 300px; margin-bottom: 15px; float:right;">
36 <h3>Carri Data</h3>
37 <center><form id="formsearch" method="POST" action="<?php echo base_url(). "c_simpanan/index" ?>">
38 <table class="tbl_fm">
39 <tr>
40 <td>Carri Berdasarkan</td>

```

Gambar 5.2 Pengkodean Aplikasi View Simpanan



```

20
21 function index()
22 {
23     if(isset($_SESSION['user']) && isset($_SESSION['divisiap']))
24     {
25         $kodestipan = trim($_GET['segment']);
26         $tglawal = $_GET['tglawal'];
27         $tglakhir = $_GET['tglakhir'];
28
29         if($tglawal != null && $tglakhir != null)
30         {
31             $tglawal = $_GET['tglawal'];
32             $tglakhir = $_GET['tglakhir'];
33         }
34         else
35         {
36             $tglawal = "2000-01-01";
37             $tglakhir = "2100-01-01";
38         }
39         $id_anggota = $_SESSION['id_anggotabecu'];
40         $keyword = $_GET['keyword'];
41
42         if(isset($keyword))
43         {
44             $keyword = $_GET['keyword'];
45             $kategori = $_GET['kategori'];
46
47             if($keyword == "selesai")
48             {
49                 $keyword = 1;
50             }
51             if($keyword == "sedang diproses")
52             {
53                 $keyword = 0;
54             }
55         }
56         if($_SESSION['user'] != "bagaswan")
57         {
58             $data['hasil'] = $this->m_simpanan->getSimpananSearchDbdivisiap($keyword, $kategori);
59         }
60         else
61         {
62             $data['hasil'] = $this->m_simpanan->getSimpananSearchDb($keyword, $id_anggota, $kategori);
63         }
64     }
65     else
66     {
67         if($_SESSION['user'] != "bagaswan")
68         {
69             $data['hasil'] = $this->m_simpanan->ambil_transaksi_simpanan($tglawal, $tglakhir);
70         }
71         else
72         {
73             $data['hasil'] = $this->m_simpanan->ambil_transaksi_simpanan($tglawal, $tglakhir, $id_anggota);
74         }
75     }
76 }

```

Gambar 5.3 Pengkodean Aplikasi Controller Display Data Simpanan



```

134
135
136
137
138
139
140
141

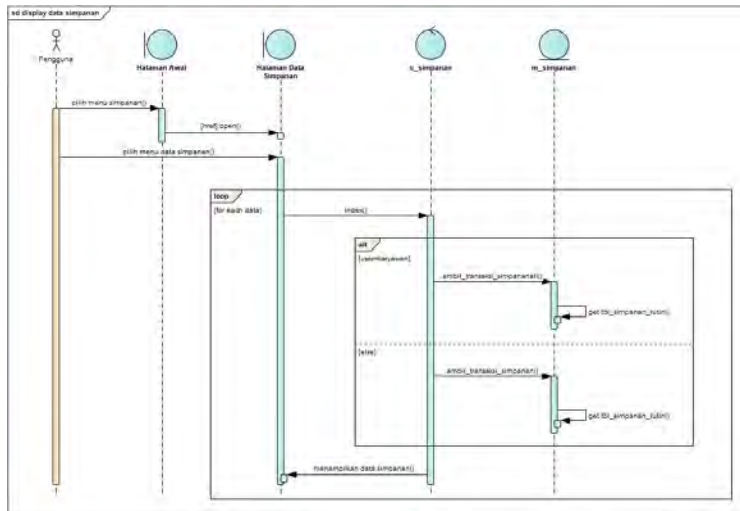
```

```

function ambil_transaksi_simpananall($tgl_awal,$tgl_akhir){
    $this->db->where('tgl_pembuatan >=', $tgl_awal);
    $this->db->where('tgl_pembuatan <=', $tgl_akhir);
    $this->db->order_by('tgl_pembuatan', 'DESC');
    $query = $this->db->get('tbl_simpanan_rutin');
    return $query;
}

```

**Gambar 5.4 Pengkodean Model Display Data Simpanan**



**Gambar 5.5 Sequence Diagram Display Data Simpanan**

## 5.2 Sampel Implementasi Aplikasi

### a. Interface Login

SILAHKAN LOGIN

username :   
 password :

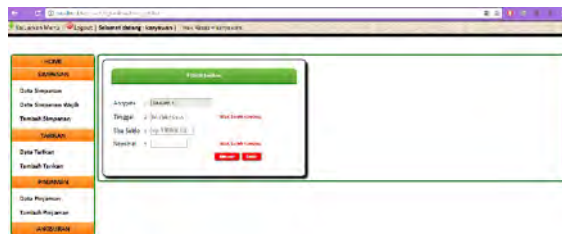
**Gambar 5.6 Interface Login**

### b. Interface form Tambah Simpanan



**Gambar 5.7 Interface Tambah Simpanan**

c. Interface form Tambah Tarikan



**Gambar 5.8 Interface Tambah Tarikan**

## 5.3 Pengujian

Pengujian dilakukan setelah pengkodean/implementasi selesai. Pengujian dilakukan berdasarkan kebutuhan fungsional. Pengujian dengan menggunakan metode ini dilakukan dengan cara menguji skenario utama dan skenario alternative pada masing-masing use case.

Berikut ini merupakan pengujian untuk masing-masing Use Case Test :

**a. Pengguna**

➤ **Test Case Login**

**Tabel 5.1 Test Case Login**

+	UCT.01.01 Login
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"><li data-bbox="400 460 1031 524">1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login</li><li data-bbox="400 527 1342 591">2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP</li><li data-bbox="400 594 1182 658">3. Pengguna menekan button “Masuk” Sistem melakukan pengecekan apakah semua field sudah diisi</li><li data-bbox="400 661 1342 725">4. Data yang dimasukkan sesuai maka pengguna langsung masuk ke halaman utama</li></ol>

+	UCT.01.01 Login				
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika username salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> <li>2. Jika password salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> <li>3. Jika username dan password tidak diisi, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> <li>4. Jika username tidak ditemukan, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> </ol>				
<b>Test Case</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Username field</b>	<b>Password field</b>	<b>Hasil</b>
	TC 01	Jika username salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”	divisisp	divisisp	Berhasil

+	UCT.01.01 Login					
	TC 02	Jika password salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”	Divisi	divisisp	Berhasil	
	TC 03	Jika username dan password tidak diisi, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”	divisisp	Divisi	Berhasil	
	TC 04	Jika username tidak ditemukan, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”	Einstein	divisisp	Berhasil	

➤ Test Case Logout

**Tabel 5.2 Test Case Logout**

+	UCT.01.02 Logout		
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login</li> <li>2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP</li> <li>3. Pengguna menekan button “Masuk” Sistem menampilkan halaman awal</li> <li>4. Aktor menekan tombol <i>Logout</i> yang terdapat pada halaman awal.</li> <li>5. Setelah aktor menekan tombol <i>Logout</i>, maka sistem akan menampilkan halaman <i>Login</i> kembali.</li> </ol>		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 05	Setelah aktor menekan tombol <i>Logout</i> , maka sistem akan menampilkan halaman <i>Login</i> kembali.	Berhasil

➤ **Test Case Display Data Simpanan**

**Tabel 5.3 Test Case Display Data Simpanan**

+	UCT.02.01 Display Data Simpanan		
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login</li> <li>2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP</li> <li>3. Pengguna menekan button “Masuk” Sistem menampilkan halaman awal</li> <li>4. Aktor memilih menu Simpanan Sistem menampilkan pilihan menu simpanan</li> <li>5. Aktor memilih menu “Data Simpanan” Sistem menampilkan keseluruhan data simpanan anggota dalam bentuk tabel</li> </ol>		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 06	Sistem menampilkan keseluruhan data simpanan anggota dalam bentuk tabel	Berhasil

➤ **Test Case Display Data Tarikan**

**Tabel 5.4 Testcase Display Data Tarikan**

+	UCT.02.02 Display Data Tarikan		
<b>Skenario Sukses Utama :</b>	1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login 2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP 3. Pengguna menekan button “Masuk” Sistem menampilkan halaman awal 4. Aktor memilih menu Tarikan Sistem menampilkan pilihan menu tarikan 5. Aktor memilih menu “Data Tarikan” Sistem menampilkan keseluruhan data tarikan anggota dalam bentuk tabel		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 07	Sistem menampilkan keseluruhan data simpanan anggota dalam bentuk tabel	Berhasil



➤ **Test Case Display Data Pinjaman**

**Tabel 5.5 Testcase Display Data Pinjaman**

+	UCT.02.03 Display Data Pinjaman		
<b>Skenario Sukses Utama :</b>	1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login 2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP 3. Pengguna menekan button “Masuk” Sistem menampilkan halaman awal 4. Aktor memilih menu Pinjaman Sistem menampilkan pilihan menu Pinjaman 5. Aktor memilih menu “Data Pinjaman” Sistem menampilkan keseluruhan data pinjaman anggota dalam bentuk tabel		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 08	Sistem menampilkan keseluruhan data simpanan anggota dalam bentuk tabel	Berhasil

➤ **Test Case Display Data Angsuran**

**Tabel 5.6 Testcase Display Data Angssruan**

+	UCT.02.04 Display Data Angsuran		
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login</li> <li>2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP</li> <li>3. Pengguna menekan button “Masuk” Sistem menampilkan halaman awal</li> <li>4. Aktor memilih menu Angsuran Sistem menampilkan pilihan menu Angsuran</li> <li>5. Aktor memilih menu “Data Angsuran” Sistem menampilkan keseluruhan data angsuran anggota dalam bentuk tabel</li> </ol>		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 09	Sistem menampilkan keseluruhan data simpanan anggota dalam bentuk tabel	Berhasil

➤ **Test Case Cari Data Simpanan**

**Tabel 5.7 Testcase Cari Data Simpanan**

+	UCT.02.05 Cari Data Simpanan		
<b>Skenario Sukses Utama :</b>	1. Aktor memilih menu Simpanan Sistem menampilkan pilihan menu Simpanan 2. Aktor memilih menu “Data Simpanan” Sistem menampilkan keseluruhan data simpanan anggota dalam bentuk tabel 3. Aktor memilih attribut filter data 4. Pengguna memasukkan keyword yang ingin dicari 5. Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 10	Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel	Berhasil

+	UCT.02.05 Cari Data Simpanan		
	TC 11	Sistem menampilkan fitur “Pengurutan Data”. Pengguna dapat mencari data transaksi simpanan yang telah disetujui maupun belum disetujui.	Berhasil

➤ **Test Case Cari Data Pinjaman**

**Tabel 5.8 Test Case Cari Data Pinjaman**

+	UCT.02.06 Cari Data Pinjaman	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Pinjaman Sistem menampilkan pilihan menu Pinjaman</li> <li>2. Aktor memilih menu “Data Pinjaman” Sistem menampilkan keseluruhan data pinjaman anggota dalam bentuk tabel</li> <li>3. Aktor memilih atribut filter data</li> <li>4. Pengguna memasukkan keyword yang ingin dicari</li> <li>5. Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel</li> </ol>	
<b>Skenario Alternatif:</b>	-	

+	UCT.02.06 Cari Data Pinjaman		
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 12	Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel	Berhasil
	TC 13	Sistem menampilkan fitur “Pengurutan Data”. Pengguna dapat mencari data transaksi pinjaman yang telah lunas maupun belum lunas.	Berhasil

➤ **Test Case Cari Data Tarikan**

**Tabel 5.9 Test Case Cari Data Tarikan**

<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Tarikan Sistem menampilkan pilihan menu Tarikan</li> <li>2. Aktor memilih menu “Data Tarikan” Sistem menampilkan keseluruhan data tarikan anggota dalam bentuk tabel</li> <li>3. Aktor memilih atribut filter data</li> <li>4. Pengguna memasukkan keyword yang ingin dicari</li> <li>5. Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel</li> </ol>
<b>Skenario Alternatif:</b>	-

Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 14	Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel	Berhasil
	TC 15	Sistem menampilkan fitur “Pengurutan Data”. Pengguna dapat mencari data transaksi tarikan yang telah disetujui maupun belum disetujui.	Berhasil

➤ **Test Case Cari Data Angsuran**

**Tabel 5.10 Test Case Cari Data Angsuran**

+	UCT.02.08 Cari Data Angsuran
	<p><b>Skenario Sukses Utama :</b></p> <ol style="list-style-type: none"> <li>1. Aktor memilih menu Angsuran Sistem menampilkan pilihan menu Angsuran</li> <li>2. Aktor memilih menu “Data Angsuran” Sistem menampilkan keseluruhan data angsuran anggota dalam bentuk tabel</li> <li>3. Aktor memilih atribut filter data</li> <li>4. Pengguna memasukkan keyword yang ingin dicari</li> </ol>

+	UCT.02.08 Cari Data Angsuran		
	5. Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 16	Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel	Berhasil

➤ **Test Case Edit Data Anggota**

**Tabel 5.11 Edit Data Anggota**

+	UCT.03.01 Edit Data Anggota		
<b>Skenario Sukses Utama :</b>	1. Aktor memilih menu Anggota Sistem menampilkan pilihan menu Anggota 2. Aktor memilih menu “Data Anggota” Sistem menampilkan keseluruhan data anggota anggota dalam bentuk tabel		

+	UCT.03.01 Edit Data Anggota		
	3. Aktor menekan icon Edit pada baris data yang ingin diubah dan memasukkan keyword yang ingin dicari Sistem menampilkan form edit data anggota 4. Aktor mengubah isi form Data Anggota Sistem menampilkan form dengan isian yang telah diubah 5. Aktor menekan tombol Simpan Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database		
<b>Skenario Alternatif:</b>	1. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data anggota?”		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 17	Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database	Berhasil
	TC 18	Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data anggota?”	Berhasil



➤ **Test Case Edit Data Simpanan**

**Tabel 5.12 Test Case Edit Data Simpanan**

+	UCT.03.02 Edit Data Simpanan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Simpanan Sistem menampilkan pilihan menu Simpanan</li> <li>2. Aktor memilih menu “Data Simpanan” Sistem menampilkan keseluruhan data anggota Simpanan dalam bentuk tabel</li> <li>3. Aktor menekan icon Edit pada baris data yang ingin diubah dan memasukkan keyword yang ingin dicari Sistem menampilkan form edit data simpanan</li> <li>4. Aktor mengubah isi form Data Simpanan Sistem menampilkan form dengan isian yang telah diubah</li> <li>5. Aktor menekan tombol Simpan Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan pengeditan?”</li> </ol>

+ UCT.03.02 Edit Data Simpanan			
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 19	Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database	Berhasil
	TC 20	Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan pengeditan?”	Berhasil

➤ **Test Case Edit Data Tarikan**

**Tabel 5.13. Test Case Edit Data Tarikan**

+ UCT.03.03 Edit Data Tarikan	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>Aktor memilih menu Tarikan Sistem menampilkan pilihan menu Tarikan</li> <li>Aktor memilih menu “Data Tarikan” Sistem menampilkan keseluruhan data anggota Tarikan dalam bentuk tabel</li> </ol>

+	UCT.03.03 Edit Data Tarikan		
	3. Aktor menekan icon Edit pada baris data yang ingin diubah dan memasukkan keyword yang ingin dicari Sistem menampilkan form edit data tarikan 4. Aktor mengubah isi form Data Tarikan Sistem menampilkan form dengan isian yang telah diubah 5. Aktor menekan tombol Simpan Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database		
<b>Skenario Alternatif:</b>	1. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data tarikan?”		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 21	Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database	Berhasil
	TC 22	Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data tarikan?”	Berhasil

➤ **Test Case Edit Data Pinjaman**

**Tabel 5.14 Test Case Edit Data Pinjaman**

+	UCT.03.04 Edit Data Pinjaman
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Pinjaman Sistem menampilkan pilihan menu Pinjaman</li> <li>2. Aktor memilih menu “Data Pinjaman” Sistem menampilkan keseluruhan data anggota Pinjaman dalam bentuk tabel</li> <li>3. Aktor menekan icon Edit pada baris data yang ingin diubah dan memasukkan keyword yang ingin dicari Sistem menampilkan form edit data pinjaman</li> <li>4. Aktor mengubah isi form Data Pinjaman Sistem menampilkan form dengan isian yang telah diubah</li> <li>5. Aktor menekan tombol Simpan Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika aktor menekan tombol reset maka data angsuran per bulan akan otomatis ter-<i>reset</i></li> </ol>

+		UCT.03.04 Edit Data Pinjaman	
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 23	Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database	Berhasil
	TC 24	Jika aktor menekan tombol reset maka data angsuran per bulan akan otomatis ter- <i>reset</i>	Berhasil

➤ **Test Case Verifikasi Data Simpanan**

**Tabel 5.15 Test Case Verifikasi Data Simpanan**

+		UCT.04.01 Verifikasi Data Simpanan	
Skenario Sukses Utama :	1.	Aktor memilih menu “Data Simpanan”	
		Sistem menampilkan keseluruhan data anggota Simpanan dalam bentuk tabel	
	2.	Aktor menekan button “ACC” pada baris data yang ingin diubah	
Skenario Alternatif:	3.	Sistem menampilkan form dengan isian yang telah verifikasi	
		-	

+ UCT.04.01 Verifikasi Data Simpanan			
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 25	Sistem menampilkan form dengan isian yang telah verifikasi	Berhasil

➤ **Test Case Display Data Pelunasan Pinjaman**

**Tabel 5.16 Test Case Display Data Pelunasan Pinjaman**

+ UCT.05.01 Display Data Pelunasan Pinjaman	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login</li> <li>2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP</li> <li>3. Pengguna menekan button “Masuk” . Sistem menampilkan halaman awal</li> <li>4. Aktor memilih menu “Data Pelunasan Pinjaman” Sistem menampilkan keseluruhan data pelunasan pinjaman anggota</li> </ol>
<b>Skenario Alternatif:</b>	-

+	UCT.05.01 Display Data Pelunasan Pinjaman		
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 26	Sistem menampilkan keseluruhan data pelunasan pinjaman anggota	Berhasil

➤ **Test Case Cari Data Pelunasan Pinjaman**

**Tabel 5.17 Test Case Cari Data Pelunasan Pinjaman**

+	UCT.05.02 Cari Data Pelunasan Pinjaman	
Skenario Sukses Utama :	<ol style="list-style-type: none"> <li>1. Aktor memilih menu “Data Pelunasan Pinjaman” Sistem menampilkan keseluruhan data pelunasan pinjaman anggota</li> <li>2. Aktor memilih atribut filter data</li> <li>3. Pengguna memasukkan keyword yang ingin dicari</li> <li>4. Sistem menampilkan rekap data sesuai dengan pilihan filter data</li> </ol>	
Skenario Alternatif:	-	

+	UCT.05.02 Cari Data Pelunasan Pinjaman		
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 27	Sistem menampilkan rekap data sesuai dengan pilihan filter data	Berhasil

➤ **Test Case Unduh Data Simpanan**

**Tabel 5.18 Test Case Unduh Data Simpanan**

+	UCT.06.01 Unduh Data Simpanan		
Skenario Sukses Utama :	1. Aktor memilih menu Simpanan Sistem menampilkan pilihan menu Simpanan 2. Aktor memilih menu “Data Simpanan” Sistem menampilkan keseluruhan data simpanan anggota 3. Aktor menekan button “Print” pada baris data yang ingin diunduh Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor		
Skenario Alternatif:	-		



+	UCT.06.01 Unduh Data Simpanan		
Test Case :	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 28	Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor	Berhasil

➤ **Test Case Unduh Data Angsuran**

**Tabel 5.19 Test Case Unduh Data Angsuran**

+	UCT.06.02 Unduh Data Angsuran	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Angsuran Sistem menampilkan pilihan menu Angsuran</li> <li>2. Aktor memilih menu “Data Angsuran” Sistem menampilkan keseluruhan data angsuran anggota</li> <li>3. Aktor menekan button “Print” pada baris data yang ingin diunduh Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor</li> </ol>	
<b>Skenario Alternatif:</b>	-	

+ UCT.06.02 Unduh Data Angsuran			
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 29	Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor	Berhasil

➤ **Test Case Unduh Data Pinjaman**

**Tabel 5.20 Test Case Unduh Data Pinjaman**

+ UCT.06.03 Unduh Data Pinjaman	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>Aktor memilih menu Pinjaman Sistem menampilkan pilihan menu Pinjaman</li> <li>Aktor memilih menu “Data Pinjaman” Sistem menampilkan keseluruhan data pinjaman anggota</li> <li>Aktor menekan button “Print” pada baris data yang ingin diunduh Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor</li> </ol>
<b>Skenario Alternatif:</b>	-

+	UCT.06.03 Unduh Data Pinjaman		
Test Case :	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 30	Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor	Berhasil

➤ **Test Case Unduh Data Tarikan**

**Tabel 5.21 Test Case Unduh Data Tarikan**

+	UCT.06.03 Unduh Data Tarikan		
<b>Skenario Sukses Utama :</b>	1. Aktor memilih menu Tarikan Sistem menampilkan pilihan menu Tarikan 2. Aktor memilih menu “Data Tarikan” Sistem menampilkan keseluruhan data tarikan anggota 3. Aktor menekan button “Print” pada baris data yang ingin diunduh Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor		
<b>Skenario Alternatif:</b>	-		

+ UCT.06.03 Unduh Data Tarikan			
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 31	Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor	Berhasil

➤ **Test Case Display Data Anggota**

**Tabel 5.22 Test Case Display Data Anggota**

+ UCT.07.01 Display Data Anggota	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Anggota Sistem menampilkan pilihan menu Data Anggota”</li> <li>2. Aktor memilih menu “Data Anggota” Sistem menampilkan keseluruhan data anggota</li> </ol>
<b>Skenario Alternatif:</b>	-

+	UCT.07.01 Display Data Anggota		
Test Case :	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 32	Sistem menampilkan keseluruhan data anggota	Berhasil

➤ **Test Case Verifikasi Data Tarikan**

**Tabel 5.23 Test Case Verifikasi Data Tarikan**

+	UCT.07.02 Verifikasi Data Tarikan	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Tarikan Sistem menampilkan pilihan menu Data Tarikan”</li> <li>2. Aktor memilih menu “Data Tarikan” Sistem menampilkan keseluruhan data anggota</li> <li>3. Aktor menekan button “ACC” pada baris data yang ingin diubah Sistem menampilkan form dengan isian yang telah verifikasi dalam bentuk tabel</li> </ol>	
<b>Skenario Alternatif:</b>	-	

+ UCT.07.02 Verifikasi Data Tarikan			
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 33	Sistem menampilkan form dengan isian yang telah verifikasi dalam bentuk tabel	Berhasil

➤ **Test Case Verifikasi Data Pinjaman**

**Tabel 5.24 Test Case Verifikasi Data Pinjaman**

+ UCT.07.03 Verifikasi Data Pinjaman	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Pinjaman Sistem menampilkan pilihan menu Data Pinjaman”</li> <li>2. Aktor memilih menu “Data Pinjaman” Sistem menampilkan keseluruhan data anggota</li> <li>3. Aktor menekan button “ACC” pada baris data yang ingin diubah Sistem menampilkan form dengan isian yang telah verifikasi dalam bentuk tabel</li> </ol>
<b>Skenario Alternatif:</b>	-

+	UCT.07.03 Verifikasi Data Pinjaman		
Test Case :	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 34	Sistem menampilkan form dengan isian yang telah verifikasi dalam bentuk tabel	Berhasil

➤ **Test Case Cari Data Anggota**

**Tabel 5.25 Test Case Cari Data Anggota**

+	UCT.07.04 Cari Data Anggota		
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Anggota Sistem menampilkan pilihan menu Data Anggota”</li> <li>2. Aktor memilih menu “Data Anggota” Sistem menampilkan keseluruhan data Anggota</li> <li>3. Aktor memilih atribut filter data</li> <li>4. Pengguna memasukkan keyword yang ingin dicari</li> <li>5. Sistem menampilkan rekap data sesuai dengan pilihan filter data</li> </ol>		

+	UCT.07.04 Cari Data Anggota		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 35	Sistem menampilkan rekap data sesuai dengan pilihan filter data	Berhasil



➤ **Test Case Display Data Simpanan Wajib**

**Tabel 5.26 Test Case Display Data Simpanan Wajib**

+	UCT.08.01 Display Data Simpanan Wajib		
Skenario Sukses Utama :	1. Aktor memilih menu Simpanan Sistem menampilkan pilihan menu Data Simpanan” 2. Aktor memilih menu “Data Simpanan Wajib” Sistem menampilkan keseluruhan data Simpanan Wajib anggota dalam bentuk tabel		
Skenario Alternatif:	-		
Test Case :	ID Test Case	Nama Skenario	Hasil
	TC 36	Sistem menampilkan keseluruhan data Simpanan Wajib anggota dalam bentuk tabel	Berhasil

➤ **Test Case Cari Data Simpanan Wajib**

**Tabel 5.27 Test Case Cari Data Simpanan Wajib**

+	UCT.09.01 Cari Data Simpanan Wajib		
<b>Skenario Sukses Utama :</b>	1. Aktor memilih menu Simpanan Wajib Sistem menampilkan pilihan menu Data Simpanan Wajib” 2. Aktor memilih menu “Data Simpanan Wajib” Sistem menampilkan keseluruhan data Simpanan Wajib anggota dalam bentuk tabel 3. Aktor memilih attribut filter data 4. Pengguna memasukkan keyword yang ingin dicari Sistem menampilkan rekap data sesuai dengan pilihan filter data		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 37	Sistem menampilkan rekap data sesuai dengan pilihan filter data	Berhasil

➤ **Test Case Update Data Simpanan Wajib**

**Tabel 5.28 Test Case Update Data Simpanan Wajib**

+	UCT.09.01 Cari Data Simpanan Wajib		
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Simpanan Wajib Sistem menampilkan pilihan menu Data Simpanan Wajib”</li> <li>2. Aktor memilih menu “Data Simpanan Wajib” Sistem menampilkan keseluruhan data Simpanan Wajib anggota dalam bentuk tabel</li> <li>3. Aktor memasukkan “Pilih Bulan” dan “Pilih Tahun”</li> <li>4. Aktor menekan tombol “Lihat” Sistem menampilkan data yang telah di filter oleh aktor</li> <li>5. Aktor menekan icon “Bayar” apabila anggota telah membayar Sistem menampilkan data dengan isian yang telah diubah dalam bentuk tabel</li> </ol>		
<b>Skenario Alternatif:</b>	-		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 38	Sistem menampilkan data dengan isian yang telah diubah dalam bentuk tabel	Berhasil

➤ **Test Case Tambah Data Simpanan**

**Tabel 5.29 Test Case Tambah Data Simpanan**

+	UCT.10.01 Tambah Data Simpanan		
<b>Skenario Sukses Utama :</b>	1. Aktor memilih menu Simpanan Sistem menampilkan pilihan menu Data Simpanan” 2. Aktor memilih menu “Tambah Simpanan” Sistem menampilkan form tambah simpanan 3. Aktor mengisi data tambah simpanan Sistem menampilkan form tambah data simpanan 4. Aktor menekan tombol Simpan 5. Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database		
<b>Skenario Alternatif:</b>	1. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!” 2. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan transaksi?”		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 39	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database	Berhasil

+	UCT.10.01 Tambah Data Simpanan		
	TC 40	Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”	Berhasil
	TC 41	Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan transaksi?”	Berhasil

➤ **Test Case Tambah Data Tarikan**

**Tabel 5.30 Test Case Tambah Data Tarikan**

+	UCT.10.02 Tambah Data Tarikan	
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>Aktor memilih menu Tarikan Sistem menampilkan pilihan menu Data Tarikan”</li> <li>Aktor memilih menu “Tambah Tarikan” Sistem menampilkan form tambah tarikan</li> <li>Aktor mengisi data tambah tarikan Sistem menampilkan form tambah data tarikan</li> <li>Aktor menekan tombol Simpan</li> <li>Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</li> </ol>	

+	UCT.10.02 Tambah Data Tarikan		
<b>Skenario Alternatif:</b>	1. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!” 2. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan transaksi?”		
<b>Test Case :</b>	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 42	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database	Berhasil
	TC 43	Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!” dan tombol “Simpan tidak dapat di klik”	Berhasil
	TC 44	Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan transaksi?”	Berhasil

➤ **Test Case Tambah Data Pinjaman**

**Tabel 5.31 Test Case Tambah Data Pinjaman**

+	UCT.10.03 Tambah Data Pinjaman
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Pinjaman Sistem menampilkan pilihan menu Data Pinjaman”</li> <li>2. Aktor memilih menu “Tambah Pinjaman” Sistem menampilkan form tambah pinjaman</li> <li>3. Aktor mengisi data tambah pinjaman Sistem menampilkan form tambah data pinjaman</li> <li>4. Aktor menekan tombol Simpan</li> <li>5. Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem akan menampilkan pemberitahuan “Anda harus membayarkan minimal 50 % dari pinjaman Anda sebelumnya!”</li> <li>2. Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem tidak menampilkan tombol simpan</li> <li>3. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”</li> </ol>

+	UCT.10.03 Tambah Data Pinjaman		
	4. Jika aktor menekan tombol reset maka sistem akan otomatis menghapus semua data yang telah diisikan didalam form		
Test Case :	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 45	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database	Berhasil
	TC 46	Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem akan menampilkan pemberitahuan “Anda harus membayarkan minimal 50 % dari pinjaman Anda sebelumnya!”	Berhasil
	TC 47	Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem tidak menampilkan tombol simpan	Berhasil
	TC 48	Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”	Berhasil
	TC 49	Jika aktor menekan tombol reset maka sistem akan otomatis menghapus semua data yang telah diisikan didalam form	Berhasil



➤ **Test Case Tambah Data Anggota**

**Tabel 5.32 Test Case Tambah Data Anggota**

+	UCT.11.01 Tambah Data Anggota
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Anggota Sistem menampilkan pilihan menu Data Anggota”</li> <li>2. Aktor memilih menu “Tambah Anggota” Sistem menampilkan form tambah anggota</li> <li>3. Aktor mengisi data tambah anggota Sistem menampilkan form tambah data anggota</li> <li>4. Aktor menekan tombol Simpan</li> <li>5. Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Harap isi bidang ini” dan terdapat notifikasi “tidak boleh kosong” pada bagian kanan form</li> <li>2. Jika aktor tidak memasukkan tanggal secara keseluruhan, maka sistem akan menampilkan pemberitahuan “Masukkan nilai yang valid. Bidang tersebut tidak lengkap atau memiliki tanggal yang tidak valid”.</li> <li>3. Nomor telepon dan gaji yang dimasukkan harus berupa angka</li> </ol>

+	UCT.11.01 Tambah Data Anggota		
	4. Apabila “Role” belum dipilih, maka sistem akan menampilkan pemberitahuan “Harap pilih bidang ini!” 5. Jika aktor menekan tombol batal maka sistem akan menampilkan halaman awal		
Test Case :	<b>ID Test Case</b>	<b>Nama Skenario</b>	<b>Hasil</b>
	TC 50	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database	Berhasil
	TC 51	Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Harap isi bidang ini” dan terdapat notifikasi “tidak boleh kosong” pada bagian kanan form	Berhasil
	TC 52	Jika aktor tidak memasukkan tanggal secara keseluruhan, maka sistem akan menampilkan pemberitahuan “Masukkan nilai yang valid. Bidang tersebut tidak lengkap atau memiliki tanggal yang tidak valid”.	Berhasil
	TC 53	Nomor telepon dan gaji yang dimasukkan harus berupa angka	Berhasil
	TC 54	Jika aktor menekan tombol batal maka sistem akan menampilkan halaman awal	Berhasil

➤ **Test Case Tambah Daftar Tagihan Angsuran**

**Tabel 5.33 Test Case Tambah Daftar Tagihan Angsuran**

+	UCT.11.01 Tambah Daftar Tagihan Angsuran								
Skenario Sukses Utama :	<div>1. Aktor memilih menu Angsuran Sistem menampilkan pilihan menu angsuran</div> <div>2. Aktor memilih menu “Tambah Daftar Tagihan Angsuran” Sistem menampilkan form tambah Daftar Tagihan Angsuran</div> <div>3. Aktor mengisi data tambah Daftar Tagihan Angsuran Sistem menampilkan form tambah Daftar Tagihan Angsuran</div> <div>4. Aktor menekan tombol “Tambahkan” Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</div>								
Skenario Alternatif:	<div>1. Jika field “Jumlah Bayar” pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”</div> <div>2. Jika aktor belum memilih “ID Pinjaman Belum Lunas” maka sistem menampilkan pesan “Lengkapi Data!”</div>								
Test Case :	<table><tr><td>ID Test Case</td><td>Nama Skenario</td><td>Hasil</td></tr><tr><td colspan="3"></td></tr></table>			ID Test Case	Nama Skenario	Hasil			
ID Test Case	Nama Skenario	Hasil							

+	UCT.11.01 Tambah Daftar Tagihan Angsuran		
	TC 55	Jika field “Jumlah Bayar” pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”	Berhasil
	TC 56	Jika aktor belum memilih “ID Pinjaman Belum Lunas” maka sistem menampilkan pesan “Lengkapi Data!”	Berhasil

➤ **Test Case Edit Akun**

**Tabel 5.34 Test Case Edit Akun**

+	UCT.01.12.01 Edit Akun
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat mengubah nama dan password penggunaa
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login</li> <li>2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP</li> <li>3. Pengguna menekan button “Masuk” Sistem menampilkan halaman awal</li> <li>4. Pengguna memilih fitur “Pengaturan Akun” Sistem menampilkan halaman pengaturan akun</li> <li>5. Pengguna merubah Nama dan Password lama dengan yang baru</li> </ol>

+	UCT.01.12.01 Edit Akun		
	6. Pengguna menekan button “Simpan” Sistem menampilkan halaman awal		
<b>Skenario Alternatif:</b>	1. Pengguna batal merubah data nama dan password dengan menekan tombol “Batal”. Sistem menampilkan alert “Apakah anda yakin untuk batal mengubah data anggota?” 2. Jika password yang diinputkan pada “Ulangi Password Baru” tidak sesuai dengan “Password Baru” yang telah dimasukkan, maka akan muncul pemberitahuan “Password yang anda masukkan tidak sesuai!”		
<b>Test Case</b>	<b>ID Skenario</b>	<b>Ekspektasi</b>	<b>Hasil</b>
	TC 57	Pengguna menekan button “Simpan”. Sistem menampilkan halaman awal	Berhasil
	TC 58	Pengguna batal merubah data nama dan password dengan menekan tombol “Batal”. Sistem menampilkan alert “Apakah anda yakin untuk batal mengubah data anggota?”	Berhasil
	TC 59	Jika password yang diinputkan pada “Ulangi Password Baru” tidak sesuai dengan “Password Baru” yang telah dimasukkan, maka akan muncul pemberitahuan “Password yang anda masukkan tidak sesuai!”	Berhasil

## BAB VI HASIL DAN PEMBAHASAN

Bagian ini menjelaskan hasil dan pembahasan dari pengembangan aplikasi koperasi simpan pinjam PT Wonosari Jaya :

### 6.1 Aplikasi Koperasi

Hasil dari rancang bangun aplikasi koperasi simpan pinjam ini yaitu berupa aplikasi yang sesuai dengan permintaan Koperasi PT Wonosari Jaya. Berikut ini merupakan beberapa contoh gambar bentuk interface aplikasi koperasi simpan pinjam :

No	ID Anggota	Nama Anggota	No. Rekening	Saldo	Aksi
1	01010201	Adi Satrio	00010201	Rp.25.000,00	[Simpanan] [Hapus]
2	01010202	Adi Satrio	00010202	Rp.25.000,00	[Simpanan] [Hapus]
3	01010203	Adi Satrio	00010203	Rp.25.000,00	[Simpanan] [Hapus]
4	01010204	Adi Satrio	00010204	Rp.25.000,00	[Simpanan] [Hapus]
5	01010205	Adi Satrio	00010205	Rp.25.000,00	[Simpanan] [Hapus]
6	01010206	Adi Satrio	00010206	Rp.25.000,00	[Simpanan] [Hapus]

**Gambar 6.1 Interface Lihat Data Simpanan**

No	ID Anggota	Nama Anggota	No. Rekening	Saldo	Aksi
1	01010201	Adi Satrio	00010201	Rp.25.000,00	[Simpanan] [Hapus]
2	01010202	Adi Satrio	00010202	Rp.25.000,00	[Simpanan] [Hapus]
3	01010203	Adi Satrio	00010203	Rp.25.000,00	[Simpanan] [Hapus]
4	01010204	Adi Satrio	00010204	Rp.25.000,00	[Simpanan] [Hapus]
5	01010205	Adi Satrio	00010205	Rp.25.000,00	[Simpanan] [Hapus]

**Gambar 6.2 Interface Transaksi Pinjaman**

Gambar 6.3 Interface Form Tambah Simpanan

No	No. Angk	Nama Anggota	No. Pinjaman	Jumlah Pinjaman	Jumlah Angsuran	Status
1	1000000	Anggota A	1000000	1000000	1000000	1000000
2	1000000	Anggota B	1000000	1000000	1000000	1000000
3	1000000	Anggota C	1000000	1000000	1000000	1000000
4	1000000	Anggota D	1000000	1000000	1000000	1000000
5	1000000	Anggota E	1000000	1000000	1000000	1000000

Gambar 6.4 Interface Form Transaksi Pinjaman

## 6.2 Validasi Aplikasi

Validasi aplikasi digunakan untuk memastikan bahwa aplikasi sesuai dengan perancangan yang sudah dibuat. Validasi ini dilakukan dengan menguji hasil perangkat lunak yang sudah dibuat dengan skenario yang sudah disusun. Adapun hasil dari pengujian terdapat pada dokumen pengujian.

## 6.3 Verifikasi Aplikasi

Verifikasi digunakan untuk memastikan bahwa hasil perancangan sesuai dengan kebutuhan dari client. Proses verifikasi dilakukan pada setiap tahapan yang digunakan dalam pembuatan perangkat lunak. Verifikasi dilakukan dengan cara membuat *acceptance checklist* yang berisikan fitur-fitur yang sudah direview dan disetujui oleh *client*. Adapun *acceptance checklist* terdapat pada bagian lampiran.

## BAB VII KESIMPULAN DAN SARAN

Pada bab ini dibahas mengenai kesimpulan dari semua proses yang telah dilakukan dan saran yang dapat diberikan untuk pengembangan yang lebih baik.

### 7.1 Kesimpulan

Berdasarkan proses-proses yang telah dilakukan dalam pengerjaan tugas akhir ini maka ada beberapa kesimpulan yang dapat diambil. Aplikasi Koperasi Karyawan yang dibuat berdasar pada metode Iconix Process yang didalamnya terdapat metode VORD yang digunakan untuk penggalian kebutuhan. Penggalian kebutuhan yang didapatkan dari wawancara kepada salah satu anggota Koperasi Karyawan PT Wonosari Jaya. Tahapan pertama yang dilakukan adalah *Requirement Analysis* dengan menghasilkan pemetaan kebutuhan koperasi berdasarkan masing-masing *viewpoint*. Setelah daftar kebutuhan koperasi terbentuk, tahap selanjutnya adalah pembuatan Domain Model, GUI Storyboard, serta Use Case Diagram dan Use Case Description yang digunakan pada tahap selanjutnya. Tahapan kedua yang dilakukan adalah *Analysis and Preliminary Design* dengan menggunakan *input* data dari tahapan yang telah dilakukan sebelumnya, sehingga menghasilkan Robustness Diagram. Tahap ketiga dalam tugas akhir ini adalah *Design & Coding* yang menghasilkan Sequence Diagram, Pengkodean dan Class Diagram. Pada tahapan pengkodean menghasilkan beberapa fungsi, diantaranya :

1. Display Data
2. Edit Data
3. Unduh Data
4. Verifikasi Data
5. Tambah Data
6. Update Data Simpanan Wajib



7. Pencarian Data Transaksi
8. Tambah Daftar Tagihan Angsuran
9. Edit Akun

Aplikasi Koperasi Karyawan yang dibuat telah diuji dengan menggunakan *test case* dan sesuai dengan kebutuhan yang telah dispesifikasikan.

## 7.2 Saran

Adapun saran yang diberikan untuk pengembangan kedepan yaitu sebagai berikut :

1. Ditambahkan *dashboard* pada masing-masing pengguna, khususnya pada karyawan dengan tujuan mengetahui data transaksi pada karyawan tersebut
2. Ditambahkan notifikasi pada karyawan sehingga memudahkan karyawan dalam pengecekan transaksi

## DAFTAR PUSTAKA

- [1] grace, “[grace.blogstudent.mb.ipb.ac.id](http://grace.blogstudent.mb.ipb.ac.id),” 2010. [Online]. Available: <http://grace.blogstudent.mb.ipb.ac.id/2010/12/23/penggunaan-sistem-informasi-di-suatu-perusahaan-untuk-menunjang-strategisnya/>. [Diakses 23 05 2017].
- [2] D. Purnama, “PENGEMBANGUNAN SISTEM INFORMASI KOPERASI TERPADU AT-TAQWA BERBASIS WEB STUDI KASUS: KOPERASI AT-TAQWA NUGRAHA”.
- [3] I. Sommerville dan P. S. , “Viewpoints: principles, problems and a practical,” dalam *Annals of Software Engineering*, 1997.
- [4] A. M. Salem, “A Practical Extension to the Viewpoints Oriented,” (*IJACSA International Journal of Advanced Computer Science and Applications*, vol. 1, 2010.
- [5] Yulianta dan P. M. , “PENGEMBANGAN APLIKASI WEB DENGAN ICONIX PROCESS DAN UML,” *Jurnal Sistem Informasi MTI-UI*, vol. 4.
- [6] I. . S. dan P. S. , “Viewpoints: Principles, Problems and a Practical Approach to Requirements Engineering”.
- [7] G. K. dan I. S. , “Requirements Engineering with Viewpoints,” 1996.
- [8] A. M. Salem, “Requirements Analysis through Viewpoints Oriented,” *International Journal of Advanced Computer Science and Applications*, vol. 1, 2010.
- [9] Fenni, “Unified Modelling Language,” [Online]. Available: [fenni.staff.gunadarma.ac.id/Unifed/Unified](http://fenni.staff.gunadarma.ac.id/Unifed/Unified).
- [10] R. S. Pressman, *Software Engineering: A Practitioner's Approach* 6th Edition, McGraw-Hill, 2005.
- [11] D. . R. Kendall Scott, *Applying Use Case Driven Object Modeling with UML: An Annotated e-Commerce Example*, Addison Wesley, 2001.
- [12] Fenni, “Unified Modelling Language,” [Online]. Available: [tp://fenni.staff.gunadarma.ac.id/Unifed/Unified](http://fenni.staff.gunadarma.ac.id/Unifed/Unified).
- [13] Tech Target, “Use Case Diagram,” 2014. [Online]. Available: <http://whatis.techtarget.com/definition/use-case-diagram>.
- [14] N. A. Rosenberg, “TATISTICAL TESTS FOR TAXONOMIC DISTINCTIVENESS FROM OBSERVATIONS OF MONOPHYLY,” 2007.
- [15] R. C. Martin, *UML for Java Programmers*, New Jersey: Prentice Hall, 2002.
- [16] S. W. “UML 2 Sequence Diagrams: An Agile Introduction,” 2014. [Online]. Available: <http://www.agilemodeling.com/artifacts/sequenceDiagram.htm>.
- [17] J. A. Schard dan M. J. Chonoles, *UML 2 for Dummies*, Hungry Minds, 2003.
- [18] “Sequence Diagram,” [Online]. Available: <http://mahergabayu.blogspot.co.id/2011/01/sequence-diagram.html> . [Diakses 19 Maret 2017].

- [19] G. S. Object-Oriented Analysis and Design Through Unified Modeling Language, Laxmi Publications, Ltd., 2010.
- [20] T. Pender, UML Bible, Wiley, 2003.
- [21] J. R. I. J. dan G. B. , “Activity Diagram,” dalam *The Unified Modeling Language Reference Manual*, California, ADDISON-WESLEY, 1998, pp. 81-84.
- [22] Nurliana, “Pengertian, Penjelasan dan Relasi,” 2014. [Online]. Available: <https://nurliana23.wordpress.com/2014/12/14/class-diagram-pengertian-penjelasan-dan-relasi/>. [Diakses 21 February 2017].
- [23] CMM Website Interaktif MCMS Joomla(CMS), Elex Media Komputindo.
- [24] Anhar, PHP & MySql Secara Otodidak, MediaKita.
- [25] 36 Menit Belajar Komputer: Php Dan Mysql, Elex Media Komputindo.
- [26] Wikipedia, [Online]. Available: <https://id.wikipedia.org/wiki/CodeIgniter>. [Diakses 2017].
- [27] Wikipedia, “Koperasi,” dalam *Wikimedia Foundation*, 2017.
- [28] A. S. , TATA KELOLA KOPERASI KREDIT ATAU KOPERASI SIMPAN PINJAM: Implementasi kebijakan koperasi simpan pinjam terhadap manajemen pengelolaan, keorganisasian dan permodalan, Andi.
- [29] M. E. dan B. N. , Manajemen sumber daya manusia, 2000.
- [30] SHAMS-UL-ARIF, K. Q. dan S. A. K. , “REQUIREMENTS ENGINEERING PROCESSES, TOOLS/TECHNOLOGIES, & METHODOLOGIES,” *International Journal of Reviews in Computing*, p. 46, 2009-2010.
- [31] L. R. Abreo dan D. R. Windle, Software Requirements Using the Unified Process: A Practical Approach, Prentice Hall Professional, 2003.
- [32] K. B. dan I. S. , Use Case Modeling, The Addison-Wesley object technology series, Addison-Wesley Professional, 2003.
- [33] P. M. “Use Case Leveling Definitions,” [Online]. Available: <https://wiki.nci.nih.gov/display/seminfra/Use+Case+Leveling+Definitions>. [Diakses February 2017].
- [34] Setya, “Konsep Dasar Sistem,” 2017.
- [35] W. R. WICAKSANA dan W. K. S. , “EVALUASI PENGUJIAN RANCANG BANGUN APLIKASI WEB INFORMASI EKSEKUTIF PADA PEMERINTAH KABUPATEN XYZ,” 2015.
- [36] P. T. UML Bible, Wiley, 2003.

## BIODATA PENULIS



**Alvisha Farrasita Istifani**, lahir 25 Januari 1996 di kota Madiun. Penulis merupakan anak pertama dari tiga bersaudara. Penulis pernah menempuh pendidikan formal di SD Kutorejo I Tuban, SMP Al Hikmah Surabaya, SMA Al Hikmah Surabaya, dan akhirnya masuk menjadi mahasiswa program sarjana jurusan Sistem Informasi Institut Teknologi Sepuluh Nopember (ITS) angkatan 2013 dan terdaftar dengan NRP 5213100057. Pada akhir masa perkuliahan di jurusan Sistem Informasi ITS, penulis memilih untuk mengerjakan tugas akhir di Laboratorium Manajemen Sistem Informasi (MSI). Penulis mengambil topik mengenai rancang bangun sistem informasi dibawah bimbingan Sholih, S.T., M.Kom., M.SA. Selama menjadi mahasiswa di jurusan Sistem Informasi, penulis aktif dalam mengikuti kegiatan organisasi jurusan yaitu HMSI. Tidak hanya itu, penulis juga aktif menjadi panitia diberbagai kegiatan kampus dan salah satunya pernah menjadi staff acara ISE 2014 dan ISE 2015. Untuk kepentingan penelitian penulis dapat dihubungi melalui e-mail: [alvishaistifani@gmail.com](mailto:alvishaistifani@gmail.com)

*“Halaman ini sengaja dikosongkan”*

## LAMPIRAN A

### A.1 Requirement Analysis

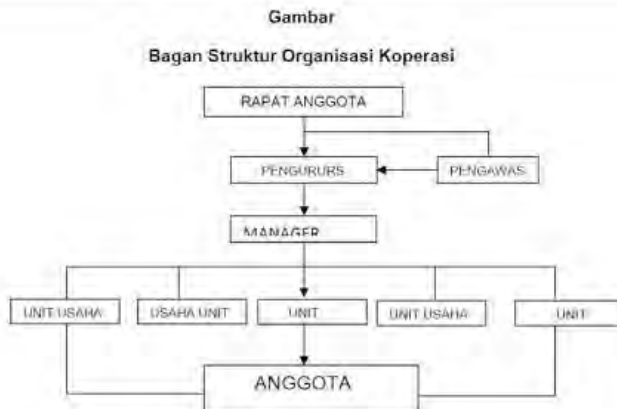
#### Hasil Wawancara

- List Pertanyaan Awal
  8. Bagaimana awal berdirinya perusahaan PT Wonosari Jaya ini?
  9. Bagaimana struktur organisasi yang ada pada koperasi perusahaan PT Wonosari Jaya?
  10. Bagaimana proses bisnis yang ada pada koperasi PT Wonosari Jaya ?
  11. Bagaimana proses simpanan pada koperasi perusahaan PT Wonosari Jaya?
  12. Bagaimana proses pinjaman pada koperasi perusahaan PT Wonosari Jaya?
  13. Apakah terdapat kendala/permasalahan yang ada pada koperasi PT Wonosari Jaya? Jika iya, apa saja kendala yang ada?
  14. Apakah harapan dari bapak terkait dengan proses transaksi pada koperasi simpan pinjam PT Wonosari Jaya pada perangkat lunak ini?
- Rekaman Wawancara

Pertanyaan dan Jawaban
<b>1. Bagaimana awal berdirinya Primkopkar PT Wonosari Jaya?</b>
Sejarah berdirinya suatu koperasi karyawan tidak bisa lepas dari perusahaan dimana karyawan tersebut bekerja. PT. WONOSARI JAYA Surabaya yang bergerak di Industri Kawat Baja yang merupakan ladang penghidupan karyawannya, sudah berdiri sejak tahun 1972 yang awal pendiriannya merupakan Industri Kawat Baja terbesar di ASEAN. Seiring dengan perkembangan jaman dan tuntutan akan adanya serikat dan berkumpul para karyawan dalam kegiatan positif sehingga pada tanggal 30 September 1982 terbentuklah salah satu organisasi sosial berazaskan

kekeluargaan dengan semangat kegotongroyongan, yaitu Koperasi Karyawan PT. WONOSARI JAYA sesuai Badan Hukum No. 5341/BH/II/82. Dalam perkembangannya Koperasi Karyawan selalu berusaha mengupdate hal-hal baru dalam pengelolaannya agar tujuan organisasi tercapai sesuai harapan.

**2. Bagaimana struktur organisasi yang ada pada Primkopkar PT Wonosari Jaya?**



**3. Bagaimana proses bisnis yang ada pada Primkopkar PT Wonosari Jaya?**

- Tugas Pengelola :
  - Merekap semua pinjaman yg disetujui dan selanjutnya dibuatkan pengesahan yg di t.tangani Ketua dan Bendahara.
  - Berdasar hasil rekap tsb.. Divisi SP meeminta uang ke Bendahara sbgai dasar pencairan uang.
- Tugas Ketua : Memverifikasi dan menandatangani data-data transaksi yang masuk
- Tugas Bagian Umum : Memverifikasi dan menandatangani data-data transaksi yang masuk
- Proses Bisnis Simpanan dan Tarikan





➤ Pengelola
➤ Ketua
➤ Bagian Umum
<b>6. Pak Bambang, saya ingin menanyakan untuk masalah Pengelola. Untuk tugas dari Pengelola apa saja ya pak? dan ada berapa petugas dari Pengelola itu sendiri?</b>
➤ Tugas Pengelola : <ul style="list-style-type: none"> <li>• Merekap semua pinjaman yg disetujui dan selanjutnya dibuatkan pengesahan yg di t.tangani Ketua dan Bendahara.</li> <li>• Berdasar hasil rekap tsb.. Divisi SP meeminta uang ke Bendahara sbgai dasar pencairan uang.</li> </ul>
<b>7. Untuk prosentase bunga pinjaman berapa persen ya pak?</b>
Kalau di Primkopkar PT. wonosari jaya <ul style="list-style-type: none"> <li>• Pinjaman Reguler bunga 1 %flat per bln</li> <li>• Pinjaman Khusus dan Pinjaman Barang bunga 1.5% per bulan.</li> </ul>
<b>8. Pak, dalam melakukan pembuatan aplikasi ini, saya membutuhkan gaji tetap tiap bulan dari masing-masing karyawan yang terlibat dalam pengembangan perangkat lunak. Bolehkah saya meminta data itu?</b>
Maaf mbak, untuk gaji tetap koperasi masih belum ada. Gaji yang diberikan sementara disama ratakan saja menjadi minimal 3.400.000 per bulan

- Interview Result

Kode Interview	Pertanyaan dan Jawaban
IR-01	<p>Sejarah berdirinya suatu koperasi karyawan tidak bisa lepas dari perusahaan dimana karyawan tersebut bekerja. PT. WONOSARI JAYA Surabaya yang bergerak di Industri Kawat Baja yang merupakan ladang penghidupan karyawannya, sudah berdiri sejak tahun 1972 yang awal pendiriannya merupakan Industri Kawat Baja terbesar di ASEAN. Seiring dengan perkembangan jaman dan tuntutan akan adanya serikat dan berkumpul para karyawan dalam kegiatan positif sehingga pada tanggal 30 September 1982 terbentuklah salah satu organisasi sosial berazaskan kekeluargaan dengan semangat kegotongroyongan, yaitu Koperasi Karyawan PT. WONOSARI JAYA sesuai Badan Hukum No. 5341/BH/II/82. Dalam perkembangannya Koperasi Karyawan selalu berusaha mengupdate hal-hal baru dalam pengelolaannya agar tujuan organisasi tercapai sesuai harapan.</p>
IR-02	<p>➤ Tugas Pengelola :</p> <ul style="list-style-type: none"> <li>• Merekap semua pinjaman yg disetujui dan selabjutnya dibuatkan pengesahan yg di t.tangani Ketua dan Bendahara.</li> <li>• Berdasar hsil rekap tsb.. Divisi SP meeminta uang ke Bendahara sbgai dasar pencairan uang.</li> </ul>

Kode Interview	Pertanyaan dan Jawaban
	<ul style="list-style-type: none"> <li>➤ Tugas Ketua : Memverifikasi dan menandatangani data-data transaksi yang masuk</li> <li>➤ Tugas Bagian Umum : Memverifikasi dan menandatangani data-data transaksi yang masuk</li> </ul>
IR-03	Terdapat 4 aktor yang dapat menggunakan aplikasi ini, diantaranya adalah karyawan, pengelola, ketua dan bagian umum.
IR-04	<p>Primkopkar PT. wonosari jaya memiliki tiga jenis pinjaman dengan bunga yang berbeda-beda :</p> <ul style="list-style-type: none"> <li>• Pinjaman Reguler bunga 1 %flat per bln</li> <li>• Pinjaman Khusus dan Pinjaman Barang bunga 1.5% per bulan</li> </ul>

- Lampiran Struktur Organisasi



**Gambar A.1 Struktur Organisasi Koperasi**

## A.2 Use Case Description

**Tabel A.1 UCD Login**

<b>+</b>	UC.01.01 Login		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melakukan <i>login</i> sehingga dapat menggunakan fasilitas-fasilitas yang sesuai dengan hak akses masing-masing.		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	-		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Pengguna memasukkan password dan username	
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika username salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> <li>2. Jika password salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> </ol>		

+	UC.01.01 Login
	<ol style="list-style-type: none"> <li>3. Jika username dan password tidak diisi, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> <li>4. Jika username tidak ditemukan, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li> </ol>

Tabel A.2 UCD Logout

+	UC.01.02 Logout
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat keluar dari sistem
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01.01 Login
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor menekan tombol <i>Logout</i> yang terdapat pada halaman awal.</li> <li>2. Setelah aktor menekan tombol <i>Logout</i>, maka sistem akan menampilkan halaman <i>Login</i> kembali.</li> </ol>
<b>Skenario Alternatif:</b>	-

Tabel A.3 UCD Display Data Simpanan

+	UC.02.01 Display Data Simpanan
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat seluruh data simpanan anggota

<b>+</b>	UC.02.01 Display Data Simpanan		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Pengguna memasukkan password dan username	
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal
	2.	Aktor memilih menu Simpanan	Sistem menampilkan pilihan menu simpanan
	3.	Aktor memilih menu “Data Simpanan”	Sistem menampilkan keseluruhan data simpanan anggota dalam bentuk tabel
<b>Skenario Alternatif :</b>	-		

**Tabel A.4 UCD Display Data Tarikan**

<b>+</b>	UC.02.02 Display Data Tarikan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat seluruh data tarikan anggota		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Pengguna memasukkan password dan username	
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal
	2.	Aktor memilih menu Tarikan	Sistem menampilkan pilihan menu tarikan
	3.	Aktor memilih menu “Data Tarikan”	Sistem menampilkan keseluruhan data tarikan anggota dalam bentuk tabel
<b>Skenario Alternatif:</b>	-		

**Tabel A.5 UCD Display Data Pinjaman**

+	UC.02.03 Display Data Pinjaman		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat seluruh data pinjaman anggota		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Pengguna memasukkan password dan username	
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal
	2.	Aktor memilih menu Pinjaman	Sistem menampilkan pilihan menu pinjaman
	3.	Aktor memilih menu “Data Pinjaman”	Sistem menampilkan keseluruhan data pinjaman anggota dalam bentuk tabel
<b>Skenario Alternatif:</b>	-		



Tabel A.6 UCD Display Data Angsuran

+	UC.02.04 Display Data Angsuran		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat seluruh data angsuran anggota		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Pengguna memasukkan password dan username	
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal
	2.	Aktor memilih menu Angsuran	Sistem menampilkan pilihan menu angsuran
	3.	Aktor memilih menu “Data Angsuran”	Sistem menampilkan keseluruhan data angsuran anggota dalam bentuk tabel
<b>Skenario Alternatif:</b>	-		

**Tabel A.7 UCD Cari Data Simpanan**

+	UC.02.05 Cari Data Simpanan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat dan mencari data Simpanan anggota		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.1 Display Data Simpanan		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Simpanan”	Sistem menampilkan keseluruhan data Simpanan anggota dalam bentuk tabel
	2.	Aktor memilih atribut filter data	
	2.1	Pengguna memasukkan keyword yang ingin dicari	
	3.		Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel
<b>Skenario Alternatif:</b>	1. Pengguna dapat memilih opsi lain dalam cari data simpanan dengan menggunakan fitur “Pengurutan Data”. Pengguna dapat mencari data transaksi simpanan yang telah disetujui maupun belum disetujui.		

**Tabel A.8 UCD Cari Data Pinjaman**

+	UC.02.06 Cari Data Pinjaman		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat dan mencari data Pinjaman anggota		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.3 Display Data Pinjaman		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Pinjaman”	Sistem menampilkan keseluruhan data Pinjaman anggota
	2.	Aktor memilih attribut filter data	
	2.1	Pengguna memasukkan keyword yang ingin dicari	
	3.		Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel
<b>Skenario Alternatif:</b>	Pengguna dapat memilih opsi lain dalam cari data pinjaman dengan menggunakan fitur “Pengurutan Data”. Pengguna dapat mencari data transaksi simpanan yang telah lunas maupun belum lunas.		

**Tabel A.9 UCD Cari Data Tarikan**

+	UCD.02.07 Cari Data Tarikan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat dan mencari data Tarikan anggota		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.2 Display Data Tarikan		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Tarikan”	Sistem menampilkan keseluruhan data Tarikan anggota
	2.	Aktor memilih atribut filter data	
	2.1	Pengguna memasukkan keyword yang ingin dicari	
	3.		Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel
<b>Skenario Alternatif:</b>	Pengguna dapat memilih opsi lain dalam cari data tarikan dengan menggunakan fitur “Pengurutan Data”. Pengguna dapat mencari data transaksi simpanan yang telah disetujui maupun belum disetujui.		

**Tabel A.10 UCD Cari Data Angsuran**

+	UCD.02.08 Cari Data Angsuran		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat dan mencari data Angsuran anggota		
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.4 Display Data Angsuran		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Angsuran”	Sistem menampilkan keseluruhan data Angsuran anggota
	2.	Aktor memilih attribut filter data	
	2.1	Pengguna memasukkan keyword yang ingin dicari	
	3.		Sistem menampilkan rekap data sesuai dengan pilihan filter data dalam bentuk tabel
<b>Skenario Alternatif:</b>	Pengguna dapat memilih opsi lain dalam cari data angsuran dengan menggunakan fitur “Pengurutan Data”. Pengguna dapat mencari data transaksi angsuran yang telah lunas maupun belum lunas.		

**Tabel A.11 UCD Edit Data Anggota**

+	UC.03.01 Edit Data Anggota		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat mengedit data anggota yang telah disimpan		
<b>Direct Aktor :</b>	Divisi SP		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC7.1 Display Data Anggota		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Anggota”	Sistem menampilkan keseluruhan data anggota
	2.	Aktor menekan icon Edit pada baris data yang ingin diubah	Sistem menampilkan form edit data anggota
	2.1	Aktor mengubah isi form Data anggota	Sistem menampilkan form dengan isian yang telah diubah
	3.	Aktor menekan tombol Simpan	Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database
<b>Skenario Alternatif:</b>	1. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data anggota?”		

Tabel A.12 UCD Edit Data Simpanan

+	UCD.03.02 Edit Data Simpanan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat mengedit data simpanan anggota yang telah disimpan		
<b>Direct Aktor :</b>	Divisi SP		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.1 Display Data Simpanan		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Simpanan”	Sistem menampilkan keseluruhan data simpanan
	2.	Aktor menekan icon Edit pada baris data yang ingin diubah	Sistem menampilkan form edit data simpanan
	2.1	Aktor mengubah isi form Data simpanan	Sistem menampilkan form dengan isian yang telah diubah
	3.	Aktor menekan tombol Simpan	Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database
<b>Skenario Alternatif:</b>	Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan pengeditan?”		

**Tabel A.13 UCD Edit Data Tarikan**

+	UCD.03.03 Edit Data Tarikan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat mengedit data tarikan anggota yang telah disimpan		
<b>Direct Aktor :</b>	Divisi SP		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.2 Display Data Tarikan		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Tarikan”	Sistem menampilkan keseluruhan data tarikan
	2.	Aktor menekan icon Edit pada baris data yang ingin diubah	Sistem menampilkan form edit data tarikan
	2.1	Aktor mengubah isi form Data tarikan	Sistem menampilkan form dengan isian yang telah diubah
	3.	Aktor menekan tombol Simpan	Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database
<b>Skenario Alternatif:</b>	Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data simpanan?”		



**Tabel A.14 UCD Edit Data Pinjaman**

+	UCD.03.04 Edit Data Pinjaman		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat mengedit data pinjaman anggota yang telah disimpan		
<b>Direct Aktor :</b>	Divisi SP		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.3 Display Data Pinjaman		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Pinjaman”	Sistem menampilkan keseluruhan data pinjaman
	2.	Aktor menekan icon Edit pada baris data yang ingin diubah	Sistem menampilkan form edit data pinjaman
	2.1	Aktor mengubah isi form Data Pinjaman	Sistem menampilkan form dengan isian yang telah diubah
	3.	Aktor menekan tombol Simpan	Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database
<b>Skenario Alternatif:</b>	Jika aktor menekan tombol reset maka data angsuran per bulan akan otomatis ter- <i>reset</i>		

**Tabel A.15 UCD Verifikasi Data Simpanan**

+	UCD.04.01 Verifikasi Data Simpanan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat melakukan verifikasi data simpanan anggota yang telah disimpan		
<b>Direct Aktor :</b>	Divisi SP		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.1 Display Data Simpanan		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Simpanan”	Sistem menampilkan keseluruhan data simpanan
	2.	Aktor menekan button “ACC” pada baris data yang ingin diubah	
	2.1		Sistem menampilkan form dengan isian yang telah verifikasi
<b>Skenario Alternatif:</b>	-		

**Tabel A.16 UCD Display Data Pelunasan Pinjaman**

+	UCD.05.01 Display Data Pelunasan Pinjaman		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat melakukan verifikasi data simpanan anggota yang telah disimpan		

+	UCD.05.01 Display Data Pelunasan Pinjaman		
Direct Aktor :	Divisi SP		
Frekuensi Penggunaan :	Bisa lebih dari satu kali		
Pre Condition :	Aktor sudah melakukan UC01.01 Login		
Skenario Sukses Utama :	No	Aktor	Sistem
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Pengguna memasukkan password dan username	
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal
	2.	Aktor memilih menu Angsuran	Sistem menampilkan pilihan menu Angsuran
	3.	Aktor memilih menu “Data Pelunasan Pinjaman”	Sistem menampilkan keseluruhan data pelunasan pinjaman anggota
Skenario Alternatif:	-		

Tabel A.17 UCD Cari Data Pelunasan Pinjaman

+	UCD.05.02 Cari Data Pelunasan Pinjaman
Ringkasan :	<i>Use-case</i> ini dimaksudkan agar aktor dapat melihat dan mencari data Angsuran anggota
Direct Aktor :	Divisi SP

+	UCD.05.02 Cari Data Pelunasan Pinjaman		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC5.1 Display Data Pelunasan Pinjaman		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu “Data Pelunasan Pinjaman”	Sistem menampilkan keseluruhan data Pelunasan Pinjaman anggota
	2.	Aktor memilih attribut filter data	
	2.1	Pengguna memasukkan keyword yang ingin dicari	
	3.		Sistem menampilkan rekap data sesuai dengan pilihan filter data
<b>Skenario Alternatif:</b>	-		

**Tabel A.18 Unduh Data Simpanan**

+	UCD.06.01 Unduh Data Simpanan
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat mengunduh data Simpanan anggota
<b>Direct Aktor :</b>	Divisi SP
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali

+	UCD.06.01 Unduh Data Simpanan		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Simpanan	Sistem menampilkan pilihan menu Simpanan
	2.	Aktor memilih menu “Data Simpanan”	Sistem menampilkan keseluruhan data Simpanan anggota
	3.	Aktor menekan <i>button “Print”</i> pada baris data yang ingin diunduh	Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor
<b>Skenario Alternatif:</b>	-		

Tabel A.19 UCD Unduh Data Angsuran

+	UCD.06.02 Unduh Data Angsuran
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat mengunduh data Angsuran anggota
<b>Direct Aktor :</b>	Divisi SP
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login

+	UCD.06.02 Unduh Data Angsuran		
Skenario Sukses Utama :	No	Aktor	Sistem
	1.	Aktor memilih menu Angsuran	Sistem menampilkan pilihan menu Angsuran
	2.	Aktor memilih menu “Data Angsuran”	Sistem menampilkan keseluruhan data Angsuran anggota
	3.	Aktor menekan <i>button “Print”</i> pada baris data yang ingin diunduh	Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor
Skenario Alternatif:	-		

Tabel A.20 UCD Unduh Data Pinjaman

+	UCD.06.03 Unduh Data Pinjaman
Ringkasan :	<i>Use-case</i> ini dimaksudkan agar pengguna dapat mengunduh data Pinjaman anggota
Direct Aktor :	Bendahara
Frekuensi Penggunaan :	Bisa lebih dari satu kali
Pre Condition :	Aktor sudah melakukan UC01.01 Login

+	UCD.06.03 Unduh Data Pinjaman		
Skenario Sukses Utama :	No	Aktor	Sistem
	1.	Aktor memilih menu Pinjaman	Sistem menampilkan pilihan menu Pinjaman
	2.	Aktor memilih menu “Data Pinjaman”	Sistem menampilkan keseluruhan data Pinjaman anggota
	3.	Aktor menekan <i>button “Print”</i> pada baris data yang ingin diunduh	Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor
<b>Skenario Alternatif:</b>	-		

Tabel A.21 UCD Unduh Data Tarikan

+	UCD.06.04 Unduh Data Tarikan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat mengunduh data Tarikan anggota		
<b>Direct Aktor :</b>	Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
Skenario Sukses Utama :	No	Aktor	Sistem
	1.	Aktor memilih menu Tarikan	Sistem menampilkan pilihan menu Tarikan

+		UCD.06.04 Unduh Data Tarikan	
	2.	Aktor memilih menu “Data Tarikan”	Sistem menampilkan keseluruhan data Tarikan anggota
	3.	Aktor menekan <i>button</i> “Print” pada baris data yang ingin diunduh	Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor
Skenario Alternatif:		-	

Tabel A.22 UCD Display Data Anggota

+		UC.07.01 Display Data Anggota	
Ringkasan :		<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat data anggota	
Direct Aktor :		Bendahara	
Frekuensi Penggunaan :		Bisa lebih dari satu kali	
Pre Condition :		Aktor sudah melakukan UC01.01 Login	
Skenario Sukses Utama :	No	Aktor	Sistem
	1.	Aktor memilih menu Anggota	Sistem menampilkan pilihan menu anggota
	2.	Aktor memilih menu “Data Anggota”	Sistem menampilkan keseluruhan data anggota



+	UC.07.01 Display Data Anggota
<b>Skenario Alternatif:</b>	-

Tabel A.23 UCD Verifikasi Data Tarikan

+	UC.07.02 Verifikasi Data Tarikan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat melakukan verifikasi data Tarikan anggota		
<b>Direct Aktor :</b>	Divisi SP, Ketua, dan Bendahara Koperasi		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Tarikan	Sistem menampilkan pilihan menu Tarikan
	2.	Aktor memilih menu “Data Tarikan”	Sistem menampilkan keseluruhan data Tarikan
	3.	Aktor menekan button “ACC” pada baris data yang ingin diubah	Sistem menampilkan form dengan isian yang telah verifikasi dalam bentuk tabel
<b>Skenario Alternatif:</b>	-		

Tabel A.24 UCD Verifikasi Data Pinjaman

+	UC.07.03 Verifikasi Data Pinjaman		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat melakukan verifikasi data Pinjaman anggota		
<b>Direct Aktor :</b>	Divisi SP, Ketua, dan Bendahara Koperasi		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor membuka halaman awal aplikasi	Sistem menampilkan halaman home
	2.	Aktor memilih menu Pinjaman	Sistem menampilkan pilihan menu Pinjaman
	3.	Aktor memilih menu “Data Pinjaman”	Sistem menampilkan keseluruhan data Pinjaman
	4.	Aktor menekan button “ACC” pada baris data yang ingin diubah	Sistem menampilkan form dengan isian yang telah verifikasi
<b>Skenario Alternatif:</b>	-		

Tabel A.25 UCD Cari Data Anggota

+	UC.07.04 Cari Data Anggota		
Ringkasan :	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat dan mencari data Anggota anggota		
Direct Aktor :	Divisi SP, Ketua, Bendahara		
Frekuensi Penggunaan :	Bisa lebih dari satu kali		
Pre Condition :	Aktor sudah melakukan UC01.01 Login		
Skenario Sukses Utama :	No	Aktor	Sistem
	1.	Aktor memilih menu Anggota	Sistem menampilkan pilihan menu Anggota
	2.	Aktor memilih menu “Data Anggota”	Sistem menampilkan keseluruhan data Anggota anggota
	3.	Aktor memilih attribut filter data	
	4.1		Pengguna memasukkan keyword yang ingin dicari
	5.		Sistem menampilkan rekap data sesuai dengan pilihan filter data
Skenario Alternatif:	-		

Tabel A.26 UCD Display Data Simpanan Wajib

+	UC.08.01 Display Data Simpanan Wajib		
Ringkasan :	Use-case ini dimaksudkan agar pengguna dapat melihat seluruh data Simpanan Wajib		
Direct Aktor :	Anggota, Bendahara		
Frekuensi Penggunaan :	Bisa lebih dari satu kali		
Pre Condition :	Aktor sudah melakukan UC01.01 Login		
Skenario Sukses Utama :	No	Aktor	Sistem
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Pengguna memasukkan password dan username	
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal
	2.	Aktor memilih menu Simpanan	Sistem menampilkan pilihan menu simpanan
	2.1	Aktor memilih menu Simpanan Wajib	Sistem menampilkan pilihan menu Simpanan Wajib
	3.	Aktor memilih menu “Data	Sistem menampilkan keseluruhan data

+	UC.08.01 Display Data Simpanan Wajib		
		Simpanan Wajib”	Simpanan Wajib anggota dalam bentuk tabel
<b>Skenario Alternatif:</b>	-		

Tabel A.27 UCD Cari Data Simpanan Wajib

+	UC.09.01 Cari Data Simpanan Wajib		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar pengguna dapat melihat dan mencari data Simpanan Wajib anggota		
<b>Direct Aktor :</b>	Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Simpanan Wajib	Sistem menampilkan pilihan menu Simpanan Wajib
	2.	Aktor memilih menu “Data Simpanan Wajib”	Sistem menampilkan keseluruhan data Simpanan Wajib anggota
	3.	Aktor memilih attribut filter data	
	3.1		Pengguna memasukkan keyword yang ingin dicari

+	UC.09.01 Cari Data Simpanan Wajib		
	4.		Sistem menampilkan rekap data sesuai dengan pilihan filter data
<b>Skenario Alternatif:</b>	-		

Tabel A.28 UCD Update Data Simpanan Wajib

+	UC.09.02 Update Data Simpanan Wajib		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat mengedit data Simpanan Wajib anggota yang telah disimpan		
<b>Direct Aktor :</b>	Bendahara		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.1 Display Data Simpanan		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Simpanan	Sistem menampilkan pilihan menu Simpanan
	2.	Aktor memilih menu “Data Simpanan Wajib”	Sistem menampilkan keseluruhan data Simpanan Wajib
	3.	Aktor memasukkan “Pilih Bulan”	

+		UC.09.02 Update Data Simpanan Wajib	
		dan “Pilih Tahun”	
	3.1	Aktor menekan tombol “Lihat”	Sistem menampilkan data yang telah di filter oleh aktor
	4.	Aktor menekan icon “Bayar” apabila anggota telah membayar	Sistem menampilkan data dengan isian yang telah diubah dalam bentuk tabel
<b>Skenario Alternatif:</b>		-	

**Tabel A.29 UCD Tambah Data Simpanan**

+		UC.10.01 Tambah Data Simpanan	
<b>Ringkasan :</b>		<i>Use-case</i> ini dimaksudkan agar aktor dapat menambah data transaksi simpanan	
<b>Direct Aktor :</b>		Anggota	
<b>Frekuensi Penggunaan :</b>		Bisa lebih dari satu kali	
<b>Pre Condition :</b>		Aktor sudah melakukan UC01.01 Login	
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Simpanan	Sistem menampilkan pilihan menu Simpanan
	2.	Aktor memilih menu “Tambah Simpanan”	Sistem menampilkan form tambah simpanan

+	UC.10.01 Tambah Data Simpanan		
	3.	Aktor mengisi data tambah simpanan	Sistem menampilkan form tambah data simpanan
	4.	Aktor menekan tombol Simpan	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”</li> <li>2. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan transaksi?”</li> </ol>		

Tabel A.30 UCD Tambah Data Tarikan

+	UCD.10.02 Tambah Data Tarikan		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat menambah data transaksi tarikan		
<b>Direct Aktor :</b>	Anggota		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.2 Display Data Tarikan		



+	UCD.10.02 Tambah Data Tarikan		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Tarikan	Sistem menampilkan pilihan menu Tarikan
	2.	Aktor memilih menu “Tambah Tarikan”	Sistem menampilkan form tambah Tarikan
	3.	Aktor mengisi data tambah tarikan	Sistem menampilkan form tambah data tarikan
	4.	Aktor menekan tombol Simpan	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database
<b>Skenario Alternatif:</b>	1. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!” dan tombol “Simpan tidak dapat di klik” 2. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan transaksi?”		

Tabel A.31 UCD Tambah Data Pinjaman

+	UC.10.03 Tambah Data Pinjaman
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat menambah data transaksi pinjaman
<b>Direct Aktor :</b>	Anggota
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali

+	UC.10.03 Tambah Data Pinjaman		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login dan UC2.3 Display Data Pinjaman		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Pinjaman	Sistem menampilkan pilihan menu Pinjaman
	2.	Aktor memilih menu “Tambah Pinjaman”	Sistem menampilkan form tambah pinjaman
	3.	Aktor mengisi data tambah pinjaman	Sistem menampilkan form tambah data pinjaman
	4.	Aktor menekan tombol Simpan	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem akan menampilkan pemberitahuan “Anda harus membayarkan minimal 50 % dari pinjaman Anda sebelumnya!”</li> <li>2. Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem tidak menampilkan tombol simpan</li> <li>3. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”</li> </ol>		

+	UC.10.03 Tambah Data Pinjaman
	4. Jika aktor menekan tombol reset maka sistem akan otomatis menghapus semua data yang telah diisikan didalam form

**Tabel A.32 UCD Tambah Data Anggota**

+	UC.11.01 Tambah Data Anggota		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat menambah data anggota		
<b>Direct Aktor :</b>	Divisi SP		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Anggota	Sistem menampilkan pilihan menu Anggota
	2.	Aktor memilih menu “Tambah Data Anggota”	Sistem menampilkan form tambah data Anggota
	3.	Aktor mengisi data tambah Anggota	Sistem menampilkan form tambah data Anggota
	4.	Aktor menekan tombol Simpan	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database

+	UC.11.01 Tambah Data Anggota
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Harap isi bidang ini” dan terdapat notifikasi “tidak boleh kosong” pada bagian kanan form</li> <li>2. Jika aktor tidak memasukkan tanggal secara keseluruhan, maka sistem akan menampilkan pemberitahuan “Masukkan nilai yang valid. Bidang tersebut tidak lengkap atau memiliki tanggal yang tidak valid”.</li> <li>3. Nomor telepon dan gaji yang dimasukkan harus berupa angka</li> <li>4. Jika aktor menekan tombol batal maka sistem akan menampilkan halaman awal</li> </ol>

Tabel A.33 UCD Tambah Daftar Tagihan Angsuran

+	UC.11.02 Tambah Daftar Tagihan Angsuran		
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat menambah data angsuran		
<b>Direct Aktor :</b>	Divisi SP		
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali		
<b>Pre Condition :</b>	Aktor sudah melakukan UC01.01 Login		
<b>Skenario Sukses Utama :</b>	No	Aktor	Sistem
	1.	Aktor memilih menu Angsuran	Sistem menampilkan pilihan menu angsuran

+	UC.11.02 Tambah Daftar Tagihan Angsuran		
	2.	Aktor memilih menu “Tambah Daftar Tagihan Angsuran”	Sistem menampilkan form tambah Daftar Tagihan Angsuran
	3.	Aktor mengisi data tambah Daftar Tagihan Angsuran	Sistem menampilkan form tambah Daftar Tagihan Angsuran
	4.	Aktor menekan tombol “Tambahkan”	Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database
<b>Skenario Alternatif:</b>	1. Jika field “Jumlah Bayar” pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!” 2. Jika aktor belum memilih “ID Pinjaman Belum Lunas” maka sistem menampilkan pesan “Lengkapi Data!”		

Tabel A.34 UCD Edit Akun

+	UC.12.01 Edit Akun
<b>Ringkasan :</b>	<i>Use-case</i> ini dimaksudkan agar aktor dapat mengubah nama dan password penggunaa
<b>Direct Aktor :</b>	Anggota, Divisi SP, Ketua, Bendahara Koperasi
<b>Frekuensi Penggunaan :</b>	Bisa lebih dari satu kali
<b>Pre Condition :</b>	Aktor sudah melakukan login

+	UC.12.01 Edit Akun		
<b>Skenario Sukses Utama :</b>			
	No	Aktor	Sistem
	1.	Pengguna mengakses aplikasi koperasi karyawan	Sistem menampilkan halaman login
	1.1	Pengguna memasukkan password dan username	
	1.2	Pengguna menekan button “Masuk”	Sistem menampilkan halaman awal
	2.	Pengguna memilih fitur “Pengaturan Akun”	Sistem menampilkan halaman pengaturan akun
	2.1	Pengguna merubah Nama dan Password lama dengan yang baru	
	2.2	Pengguna menekan button “Simpan”	Sistem menampilkan halaman awal
<b>Skenario Alternatif:</b>	<p>1 Pengguna batal merubah data nama dan password dengan menekan tombol “Batal”. Sistem menampilkan alert “Apakah anda yakin untuk batal mengubah data anggota?”</p> <p>2. Jika password yang diinputkan pada “Ulangi Password Baru” tidak sesuai dengan “Password Baru” yang telah dimasukkan, maka akan muncul pemberitahuan “Password yang anda masukkan tidak sesuai!”</p>		



## LAMPIRAN B

### B.1 Robustness Diagram Sistem Informasi Koperasi PT Wonosari Jaya

Tabel B.1 RB Login

+	RB.01.01.01 Login
Deskripsi :	Robustness Diagram <b>Login</b> berdasarkan <i>Use Case Diagram Login</i>
Skenario Sukses Utama :	<ol style="list-style-type: none"><li>1. Pengguna mengakses aplikasi koperasi karyawan Sistem menampilkan halaman login</li><li>2. Pengguna mengisi username dan password sesuai yang diberikan oleh Divisi SP</li><li>3. Pengguna menekan button “Masuk” Sistem melakukan pengecekan apakah semua field sudah diisi</li><li>4. Data yang dimasukkan sesuai maka pengguna langsung masuk ke halaman utama</li></ol>
Skenario Alternatif:	<ol style="list-style-type: none"><li>1. Jika username salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li><li>2. Jika password salah, maka sistem akan menampilkan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</li></ol>



+	RB.01.01.01 Login
Deskripsi :	Robustness Diagram <b>Login</b> berdasarkan <i>Use Case Diagram Login</i>
	<p>3. Jika username dan password tidak diisi, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</p> <p>4. Jika username tidak ditemukan, maka sistem akan memberikan pemberitahuan “Username dan Password yang anda masukkan tidak sesuai!”</p>
Diagram Robustness :	<pre> sequenceDiagram     actor Pengguna as Pengguna (from Model)     participant HalamanLogin as Halaman Login     participant MenampilkanHalamanLogin as Menampilkan Halaman Login     participant MenampilkanAlert as Menampilkan Alert     participant VerifikasiAkun as Verifikasi Akun     participant ApakahPasswordBenar as Apakah password sudah benar?     participant HalamanAwal as Halaman Awal     participant akun as akun      Pengguna-&gt;&gt;HalamanLogin     HalamanLogin-&gt;&gt;MenampilkanHalamanLogin     MenampilkanHalamanLogin-&gt;&gt;MenampilkanAlert     MenampilkanAlert-&gt;&gt;VerifikasiAkun     VerifikasiAkun-&gt;&gt;ApakahPasswordBenar     ApakahPasswordBenar-&gt;&gt;MenampilkanAlert : +salah     ApakahPasswordBenar-&gt;&gt;HalamanAwal : +benar     MenampilkanAlert-&gt;&gt;HalamanAwal     akun-&gt;&gt;VerifikasiAkun   </pre>

Tabel B.2 RB Logout

+	RB.01.01.02 Logout
<b>Deskripsi :</b>	Robustness Diagram <b>Logout</b> berdasarkan <i>Use Case Diagram Logout</i>
<b>Skenario Sukses Utama :</b>	5. Aktor menekan tombol <i>Logout</i> yang terdapat pada halaman awal. 6. Setelah aktor menekan tombol <i>Logout</i> , maka sistem akan menampilkan halaman <i>Login</i> kembali.
<b>Skenario Alternatif:</b>	-
<b>Diagram Robustness :</b>	<pre> graph TD     Pengguna[Pengguna (from Model)] --- HalamanLogin[Halaman Login]     HalamanLogin --&gt; Masukkan[Masukkan Username &amp; Password]     Masukkan --&gt; Verifikasi[Verifikasi Akun]     Verifikasi --&gt; Apakah[Apakah password sudah benar?]     Apakah -- "+salah" --&gt; Masukkan     Apakah -- "+benar" --&gt; HalamanHome[Halaman Home]     HalamanHome -- Logout --&gt; Menampilkan[Menampilkan Halaman Login]     Menampilkan --&gt; HalamanLogin   </pre>

**Tabel B.3 RB Display Data Simpanan**

<b>+</b>	RB.01.02.01 Display Data Simpanan
<b>Deskripsi :</b>	Robustness Diagram <b>Display Data Simpanan</b> berdasarkan <i>Use Case Diagram Logout</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"><li>1. Aktor membuka halaman awal aplikasi dan Sistem menampilkan halaman home</li><li>2. Pengguna masuk ke halaman awal</li><li>1. Aktor memilih menu Simpanan dan Sistem menampilkan pilihan menu simpanan</li><li>2. Aktor memilih menu “Data Simpanan” dan Sistem menampilkan keseluruhan data simpanan anggota</li></ol>
<b>Skenario Alternatif:</b>	-
<b>Diagram Robustness :</b>	<p>The diagram illustrates the robustness of the 'Display Data Simpanan' process. It starts with an actor 'Pengguna (from Modul)' interacting with a boundary 'Halaman Awal'. A use case 'Tampilkan Halaman Awal' is associated with this boundary. From 'Halaman Awal', the flow goes to a control 'Pilih "Simpanan"', which then leads to a boundary 'Halaman Simpanan'. From 'Halaman Simpanan', the flow goes to a control 'Pilih Data Simpanan', which leads to a boundary 'Halaman Data Simpanan'. Finally, from 'Halaman Data Simpanan', the flow goes to a use case 'Simpanan'.</p>

Tabel B.4 RB Display Data Tarikan

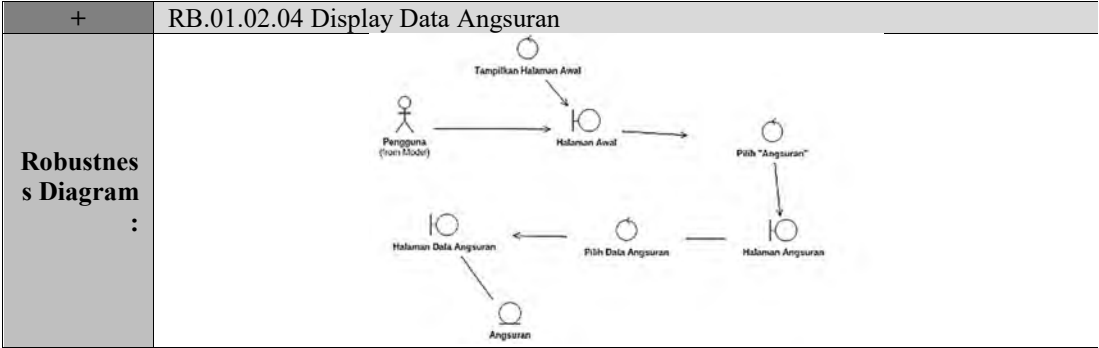
+	RB.01.02.02 Display Data Tarikan
<b>Deskripsi :</b>	Robustness Diagram <b>Display Data Tarikan</b> berdasarkan <i>Use Case Diagram Display Data Tarikan</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Pengguna masuk ke halaman awal</li> <li>3. Aktor memilih menu Tarikan dan Sistem menampilkan pilihan menu tarikan</li> <li>4. Aktor memilih menu “Data Tarikan” dan Sistem menampilkan keseluruhan data tarikan anggota</li> </ol>
<b>Skenario Alternatif:</b>	-
<b>Robustness Diagram :</b>	<pre> graph TD     Actor[Pengguna (from Mode)] --&gt; Tampilkan Halaman Awal  H1[Halaman Awal]     H1 --&gt; Pilih "Tarikan"  H2[Halaman Tarikan]     H2 --&gt; Pilih Data Tarikan  H3[Halaman Data Tarikan]     H3 --&gt; Tarikan  T[Tarikan]   </pre>

Tabel B.5 RB Display Data Pinjaman

+	RB.01.02.03 Display Data Pinjaman
Deskripsi :	Robustness Diagram <b>Display Data Pinjaman</b> berdasarkan <i>Use Case Diagram Display Data Pinjaman</i>
Skenario Sukses Utama :	<div>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</div> <div>2. Pengguna masuk ke halaman awal</div> <div>3. Aktor memilih menu Pinjaman dan Sistem menampilkan pilihan menu pinjaman</div> <div>4. Aktor memilih menu “Data Pinjaman” dan Sistem menampilkan keseluruhan data pinjaman anggota</div>
Skenario Alternatif:	-
Robustness Diagram :	<p>The Robustness Diagram illustrates the sequence of events for displaying loan data. It begins with an actor, 'Pengguna (from Model)', who triggers the process. The first event is 'Tampilkan Halaman Awal', which leads to the 'Halaman Awal' boundary. From there, the flow continues to the 'Pilih "Pinjaman"' event, which leads to the 'Halaman Pinjaman' boundary. Within this boundary, the 'Pilih Data Pinjaman' event occurs, leading to the 'Halaman Data Pinjaman' boundary. Finally, the 'Pinjaman' event is triggered, which leads to the 'Pinjaman' entity. The diagram uses standard UML notation: actors as stick figures, boundaries as rectangles with a semi-circle, and events as circles.</p>

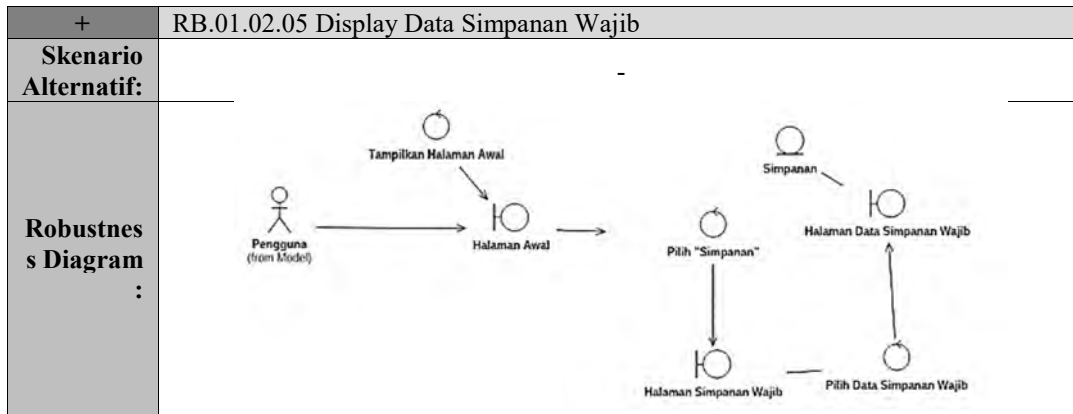
**Tabel B.6 RB Display Data Angsuran**

+	RB.01.02.04 Display Data Angsuran
<b>Deskripsi :</b>	Robustness Diagram <b>Display Data Angsuran</b> berdasarkan <i>Use Case Diagram Display Data Angsuran</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Pengguna masuk ke halaman awal</li> <li>3. Aktor memilih menu Angsuran dan Sistem menampilkan pilihan menu Angsuran</li> <li>4. Aktor memilih menu “Data Angsuran” dan Sistem menampilkan keseluruhan data Angsuran anggota</li> </ol>
<b>Skenario Alternatif:</b>	-



Tabel B.7 RB Display Data Simpanan Wajib

+	RB.01.02.05 Display Data Simpanan Wajib
Deskripsi :	Robustness Diagram <b>Display Data Simpanan Wajib</b> berdasarkan <i>Use Case Diagram Display Data Simpanan Wajib</i>
Skenario Sukses Utama :	<div>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</div> <div>2. Pengguna masuk ke halaman awal</div> <div>3. Aktor memilih menu Simpanan dan Sistem menampilkan pilihan menu Simpanan</div> <div>4. Aktor memilih menu “Data Simpanan Wajib” dan Sistem menampilkan keseluruhan data Simpanan Wajib anggota</div>

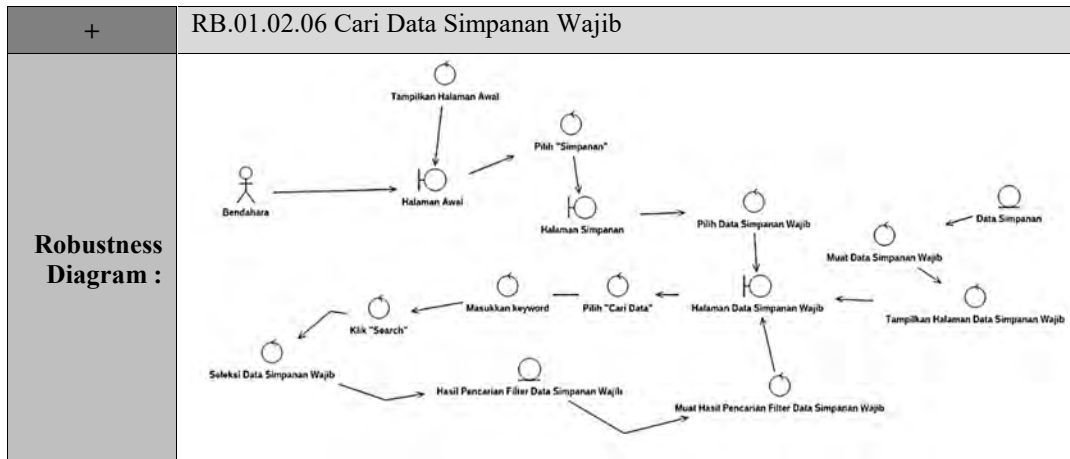


Tabel B.8 RB Cari Data Simpanan Wajib

+	RB.01.02.06 Cari Data Simpanan Wajib
Deskripsi :	Robustness Diagram <b>Cari Data Simpanan Wajib</b> berdasarkan <i>Use Case Diagram Cari Data Simpanan Wajib</i>



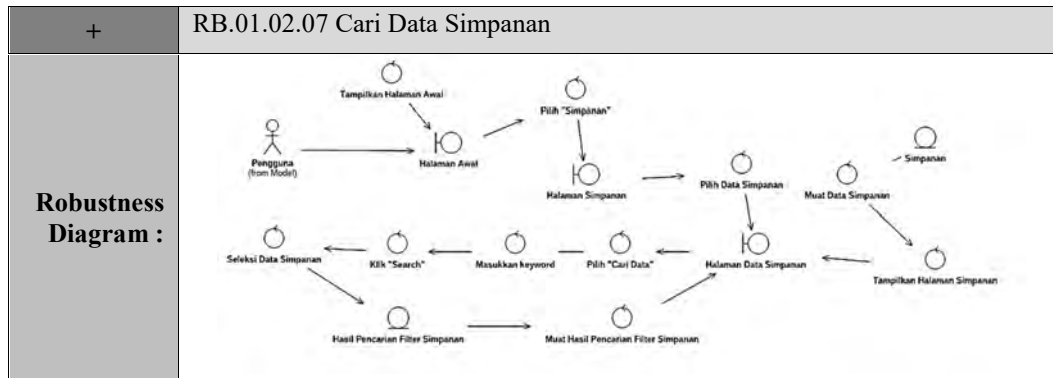
+	RB.01.02.06 Cari Data Simpanan Wajib
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Pengguna masuk ke halaman awal</li> <li>3. Aktor memilih menu Simpanan dan Sistem menampilkan pilihan menu Simpanan</li> <li>4. Aktor memilih menu “Data Simpanan Wajib” dan Sistem menampilkan keseluruhan data Simpanan Wajib anggota</li> <li>5. Aktor memilih attribut filter data lalu memasukkan keyword yang ingin dicari</li> <li>6. Sistem menampilkan rekap data sesuai dengan pilihan filter data yang telah dimasukkan</li> </ol>
<b>Skenario Alternatif:</b>	-



Tabel B.9 RB Cari Data Simpanan

+	RB.01.02.07 Cari Data Simpanan
<b>Deskripsi :</b>	Robustness Diagram <b>Cari Data Simpanan</b> berdasarkan <i>Use Case Diagram Cari Data Simpanan</i>

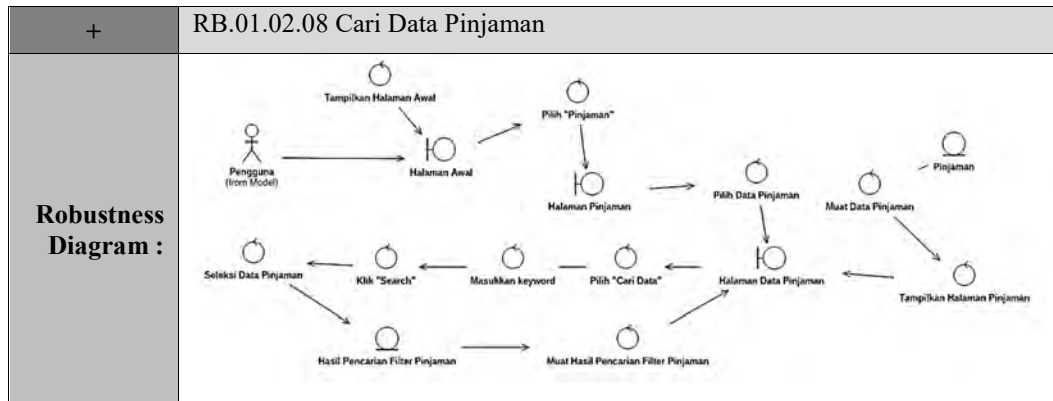
+	RB.01.02.07 Cari Data Simpanan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Pengguna masuk ke halaman awal</li> <li>3. Aktor memilih menu Simpanan dan Sistem menampilkan pilihan menu Simpanan</li> <li>4. Aktor memilih menu “Data Simpanan” dan Sistem menampilkan keseluruhan data Simpanan anggota</li> <li>5. Aktor memilih atribut filter data lalu memasukkan keyword yang ingin dicari</li> <li>6. Sistem menampilkan rekap data sesuai dengan pilihan filter data yang telah dimasukkan</li> </ol>
<b>Skenario Alternatif:</b>	-



Tabel B.10 RB Cari Data Pinjaman

+	RB.01.02.08 Cari Data Pinjaman
<b>Deskripsi :</b>	Robustness Diagram <b>Cari Data Pinjaman</b> berdasarkan <i>Use Case Diagram Cari Data Pinjaman</i>

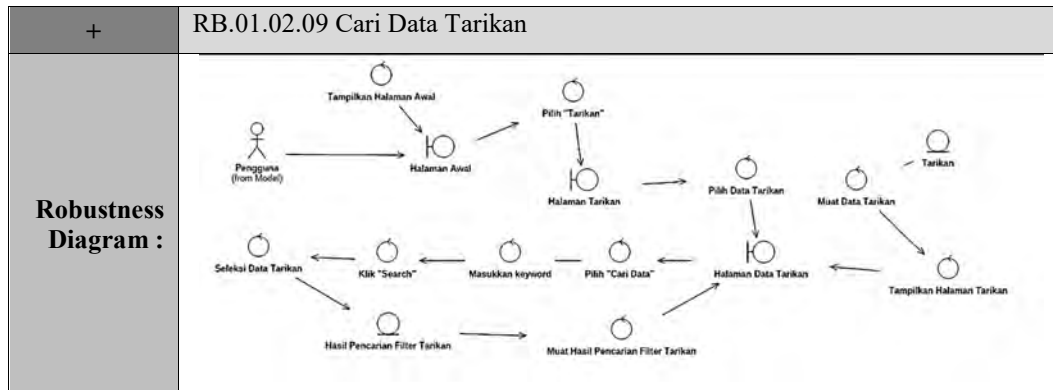
+	RB.01.02.08 Cari Data Pinjaman
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Pengguna masuk ke halaman awal</li> <li>3. Aktor memilih menu Pinjaman dan Sistem menampilkan pilihan menu Pinjaman</li> <li>4. Aktor memilih menu “Data Pinjaman” dan Sistem menampilkan keseluruhan data Pinjaman anggota</li> <li>5. Aktor memilih atribut filter data lalu memasukkan keyword yang ingin dicari</li> <li>6. Sistem menampilkan rekap data sesuai dengan pilihan filter data yang telah dimasukkan</li> </ol>
<b>Skenario Alternatif:</b>	-



Tabel B.11 RB Cari Data Tarikan

+	RB.01.02.09 Cari Data Tarikan
<b>Deskripsi :</b>	Robustness Diagram <b>Cari Data Tarikan</b> berdasarkan <i>Use Case Diagram Cari Data Tarikan</i>

+	RB.01.02.09 Cari Data Tarikan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Pengguna masuk ke halaman awal</li> <li>3. Aktor memilih menu Tarikan dan Sistem menampilkan pilihan menu Tarikan</li> <li>4. Aktor memilih menu “Data Tarikan” dan Sistem menampilkan keseluruhan data Tarikan anggota</li> <li>5. Aktor memilih atribut filter data lalu memasukkan keyword yang ingin dicari</li> <li>6. Sistem menampilkan rekap data sesuai dengan pilihan filter data yang telah dimasukkan</li> </ol>
<b>Skenario Alternatif:</b>	-

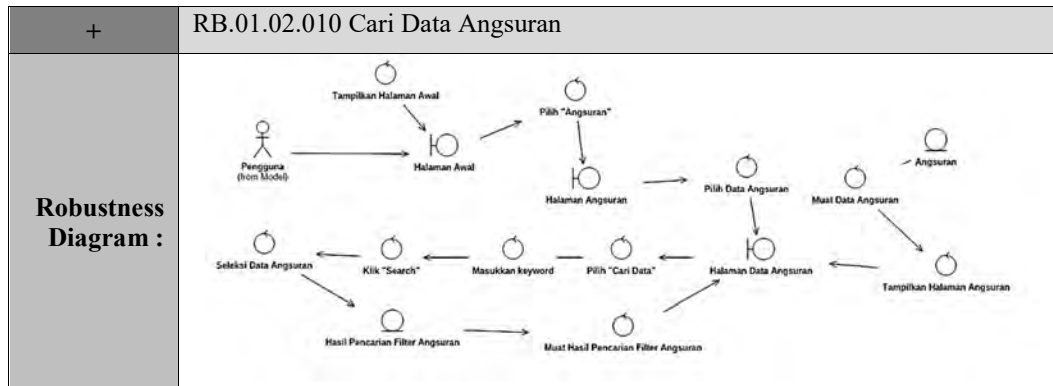


Tabel B.12 RB Cari Data Angsuran

+	RB.01.02.010 Cari Data Angsuran
<b>Deskripsi :</b>	Robustness Diagram <b>Cari Data Angsuran</b> berdasarkan <i>Use Case Diagram Cari Data Angsuran</i>



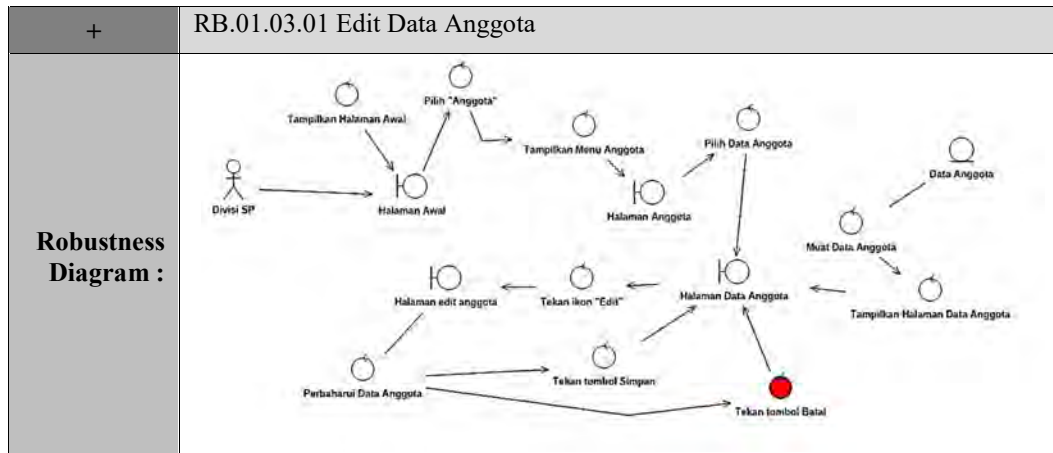
+	RB.01.02.010 Cari Data Angsuran
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Pengguna masuk ke halaman awal</li> <li>3. Aktor memilih menu Angsuran dan Sistem menampilkan pilihan menu Angsuran</li> <li>4. Aktor memilih menu “Data Angsuran” dan Sistem menampilkan keseluruhan data Angsuran anggota</li> <li>5. Aktor memilih atribut filter data lalu memasukkan keyword yang ingin dicari</li> <li>6. Sistem menampilkan rekap data sesuai dengan pilihan filter data yang telah dimasukkan</li> </ol>
<b>Skenario Alternatif:</b>	-



Tabel B.13 RB Edit Data Anggota

+	RB.01.03.01 Edit Data Anggota
<b>Deskripsi :</b>	Robustness Diagram <b>Edit Data Anggota</b> berdasarkan <i>Use Case Diagram Edit Data Anggota</i>

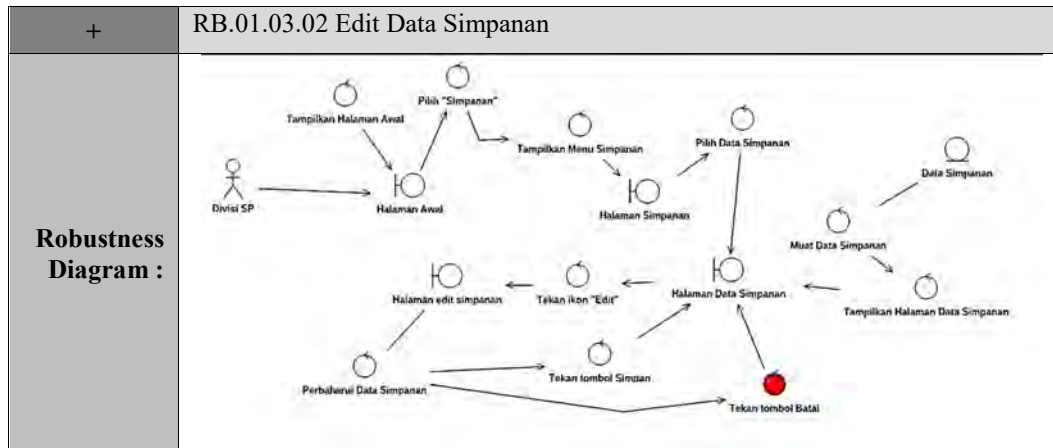
+	RB.01.03.01 Edit Data Anggota
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Anggota dan Sistem menampilkan pilihan menu Anggota</li> <li>4. Aktor memilih menu “Data Anggota” dan Sistem menampilkan keseluruhan data Anggota</li> <li>5. Aktor menekan icon Edit pada baris data yang ingin diubah dan Sistem menampilkan form edit data anggota</li> <li>6. Aktor mengubah isi form Data anggota dan Sistem menampilkan form dengan isian yang telah diubah</li> <li>7. Aktor menekan tombol Simpan dan Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<p>Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data anggota?” lalu kembali ke halaman data anggota</p>



Tabel B.14 RB Edit Data Simpanan

+	RB.01.03.02 Edit Data Simpanan
<b>Deskripsi :</b>	Robustness Diagram <b>Edit Data Simpanan</b> berdasarkan <i>Use Case Diagram Edit Data Simpanan</i>

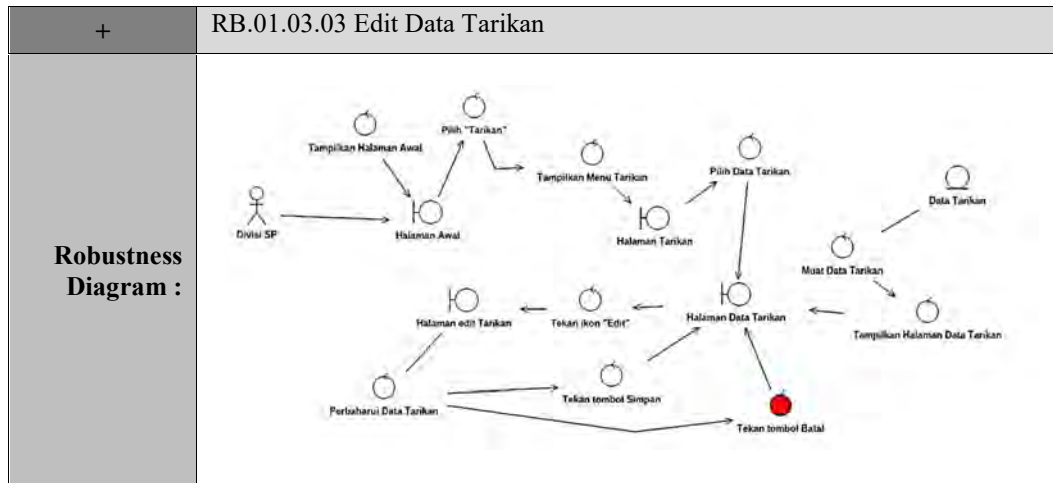
+	RB.01.03.02 Edit Data Simpanan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Simpanan dan Sistem menampilkan pilihan menu Simpanan</li> <li>4. Aktor memilih menu “Data Simpanan” dan Sistem menampilkan keseluruhan data Simpanan anggota</li> <li>5. Aktor menekan icon Edit pada baris data yang ingin diubah dan Sistem menampilkan form edit data simpanan</li> <li>6. Aktor mengubah isi form Data Simpanan dan Sistem menampilkan form dengan isian yang telah diubah</li> <li>7. Aktor menekan tombol Simpan dan Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<p>Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data simpanan?” lalu kembali ke halaman data simpanan</p>



Tabel B.15 RB Edit Data Tarikan

+	RB.01.03.03 Edit Data Tarikan
<b>Deskripsi :</b>	Robustness Diagram <b>Edit Data Tarikan</b> berdasarkan <i>Use Case Diagram Edit Data Tarikan</i>

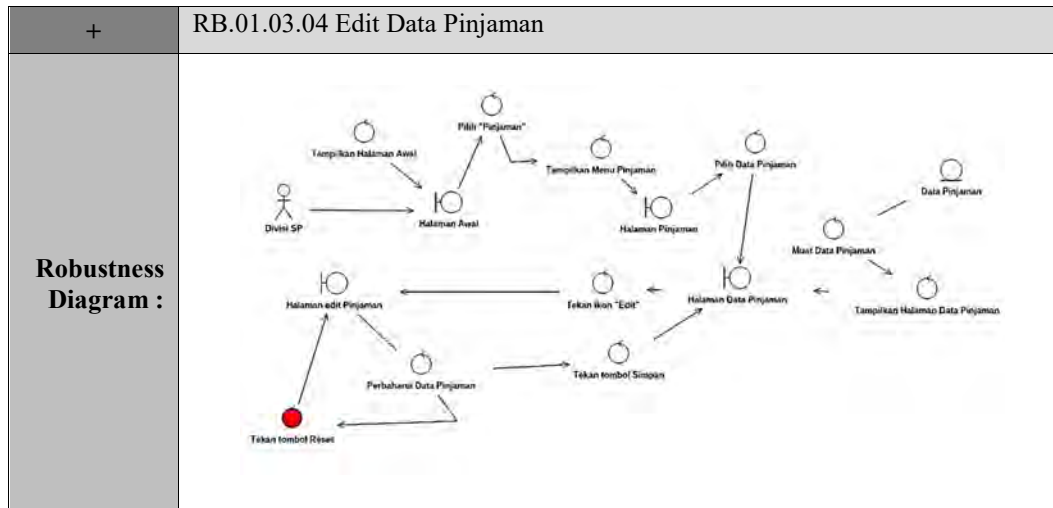
+	RB.01.03.03 Edit Data Tarikan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Tarikan dan Sistem menampilkan pilihan menu Tarikan</li> <li>4. Aktor memilih menu “Data Tarikan” dan Sistem menampilkan keseluruhan data Tarikan anggota</li> <li>5. Aktor menekan icon Edit pada baris data yang ingin diubah dan Sistem menampilkan form edit data tarikan</li> <li>6. Aktor mengubah isi form Data Tarikan dan Sistem menampilkan form dengan isian yang telah diubah</li> <li>7. Aktor menekan tombol Simpan dan Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk batal mengubah data tarikan?” lalu kembali ke halaman data tarikan





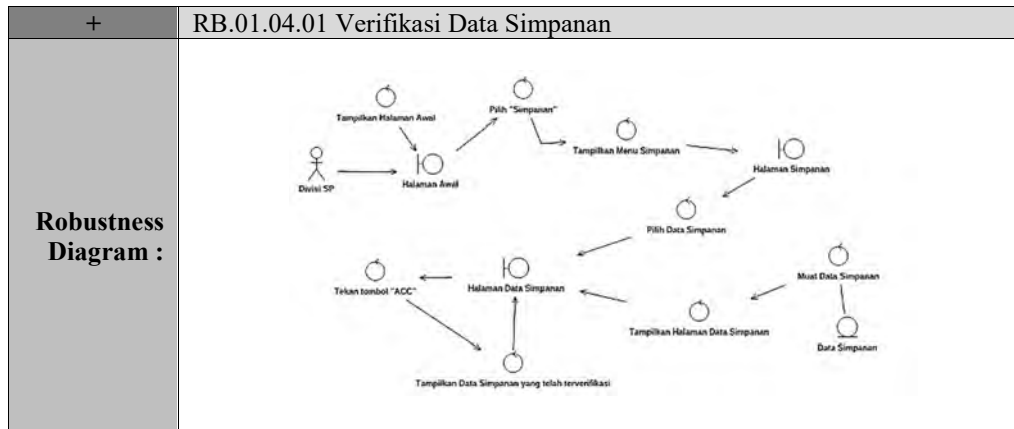
**Tabel B.16 RB Edit Data Pinjaman**

+	RB.01.03.04 Edit Data Pinjaman
<b>Deskripsi :</b>	Robustness Diagram <b>Edit Data Pinjaman</b> berdasarkan <i>Use Case Diagram Edit Data Pinjaman</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Pinjaman dan Sistem menampilkan pilihan menu Pinjaman</li> <li>4. Aktor memilih menu “Data Pinjaman” dan Sistem menampilkan keseluruhan data Pinjaman anggota</li> <li>5. Aktor menekan icon Edit pada baris data yang ingin diubah dan Sistem menampilkan form edit data pinjaman</li> <li>6. Aktor mengubah isi form Data Pinjaman dan Sistem menampilkan form dengan isian yang telah diubah</li> <li>7. Aktor menekan tombol Simpan dan Sistem menampilkan notifikasi sukses dan menyimpan perubahan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	Jika aktor menekan tombol reset maka data angsuran per bulan akan otomatis ter-reset



**Tabel B.17 RB Verifikasi Data Simpanan**

+	RB.01.04.01 Verifikasi Data Simpanan
<b>Deskripsi :</b>	Robustness Diagram <b>Verifikasi Data Simpanan</b> berdasarkan <i>Use Case Diagram Verifikasi Data Simpanan</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Simpanan dan Sistem menampilkan pilihan menu Simpanan</li> <li>4. Aktor memilih menu “Data Simpanan” dan Sistem menampilkan keseluruhan data Simpanan anggota</li> <li>5. Aktor menekan button “ACC” pada baris data yang ingin diubah</li> <li>6. Sistem menampilkan form dengan isian yang telah verifikasi</li> </ol>
<b>Skenario Alternatif:</b>	-



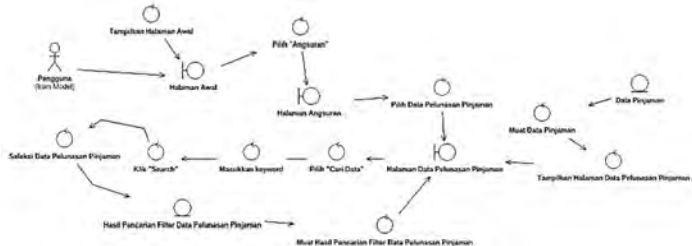
Tabel B.18 RB Display Data Pelunasan Pinjaman

+	RB.01.05.01 Display Data Pelunasan Pinjaman
<b>Deskripsi :</b>	Robustness Diagram <b>Display Data Pelunasan Pinjaman</b> berdasarkan <i>Use Case Diagram Display Data Pelunasan Pinjaman</i>

+	RB.01.05.01 Display Data Pelunasan Pinjaman
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Angsuran dan Sistem menampilkan pilihan menu Angsuran</li> <li>4. Aktor memilih menu “Data Pelunasan Pinjaman” dan Sistem menampilkan keseluruhan data Data Pelunasan Pinjaman anggota</li> </ol>
<b>Skenario Alternatif:</b>	-
<b>Robustness Diagram :</b>	<pre> sequenceDiagram     actor Divisi SP     participant HA as Halaman Awal     participant PA as Pilih "Angsuran"     participant TM as Tampilkan Menu Angsuran     participant HA2 as Halaman Angsuran     participant PDP as Pilih Data Pelunasan Pinjaman     participant HDP as Halaman Data Pelunasan Pinjaman     participant MDP as Muat Data Pelunasan Pinjaman     participant DP as Data Pinjaman     participant THDP as Tampilkan Halaman Data Pelunasan Pinjaman      Divisi SP-&gt;&gt;HA:      activate HA     HA-&gt;&gt;PA:      deactivate HA     activate PA     PA-&gt;&gt;TM:      deactivate PA     activate TM     TM-&gt;&gt;HA2:      deactivate TM     activate HA2     HA2-&gt;&gt;PDP:      deactivate HA2     activate PDP     PDP-&gt;&gt;HDP:      deactivate PDP     activate HDP     HDP-&gt;&gt;MDP:      deactivate HDP     activate MDP     MDP-&gt;&gt;DP:      deactivate MDP     activate DP     DP-&gt;&gt;THDP:      deactivate DP     deactivate THDP   </pre>

**Tabel B.19 RB Cari Data Pelunasan Pinjaman**

+	RB.01.05.02 Cari Data Pelunasan Pinjaman
<b>Deskripsi :</b>	Robustness Diagram <b>Cari Data Pelunasan Pinjaman</b> berdasarkan <i>Use Case Diagram Cari Data Pelunasan Pinjaman</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Angsuran dan Sistem menampilkan pilihan menu Angsuran</li> <li>4. Aktor memilih menu “Data Pelunasan Pinjaman” dan Sistem menampilkan keseluruhan data Data Pelunasan Pinjaman anggota</li> <li>5. Aktor memilih atribut filter data</li> <li>6. Pengguna memasukkan keyword yang ingin dicari</li> <li>7. Sistem menampilkan rekap data sesuai dengan pilihan filter data</li> </ol>
<b>Skenario Alternatif:</b>	-

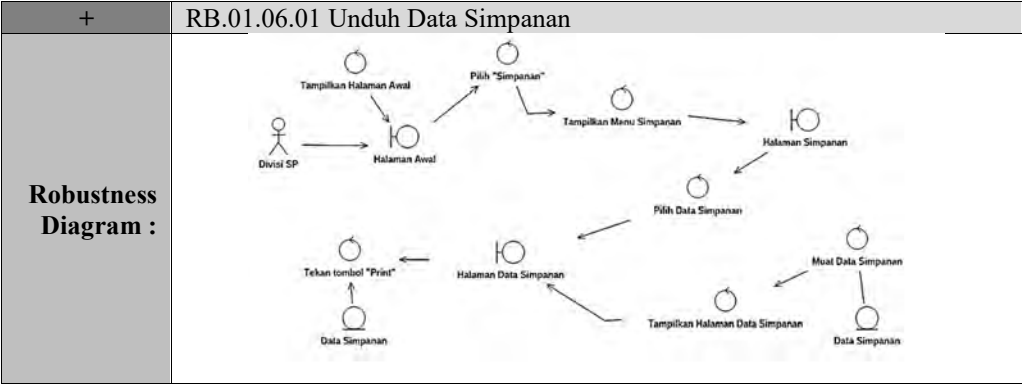
+	RB.01.05.02 Cari Data Pelunasan Pinjaman
Robustness Diagram :	

Tabel B.20 RB Unduh Data Simpanan

+	RB.01.06.01 Unduh Data Simpanan
Deskripsi :	Robustness Diagram Unduh Data Simpanan berdasarkan <i>Use Case Diagram Unduh Data Simpanan</i>

+	RB.01.06.01 Unduh Data Simpanan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Simpanan dan Sistem menampilkan pilihan menu Simpanan</li> <li>4. Aktor memilih menu “Data Simpanan” dan Sistem menampilkan keseluruhan data Data Simpanan anggota</li> <li>5. Aktor menekan button “Print” pada baris data yang ingin diunduh, lalu Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor</li> </ol>
<b>Skenario Alternatif:</b>	-

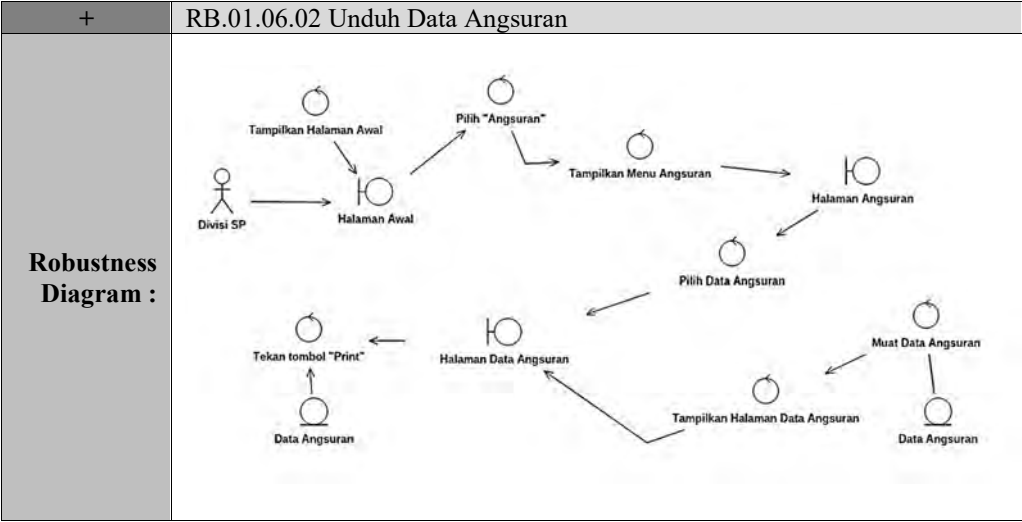




**Tabel B.21 RB Unduh Data Angsuran**

+	RB.01.06.02 Unduh Data Angsuran
<b>Deskripsi :</b>	Robustness Diagram <b>Unduh Data Angsuran</b> berdasarkan <i>Use Case Diagram Unduh Data Angsuran</i>

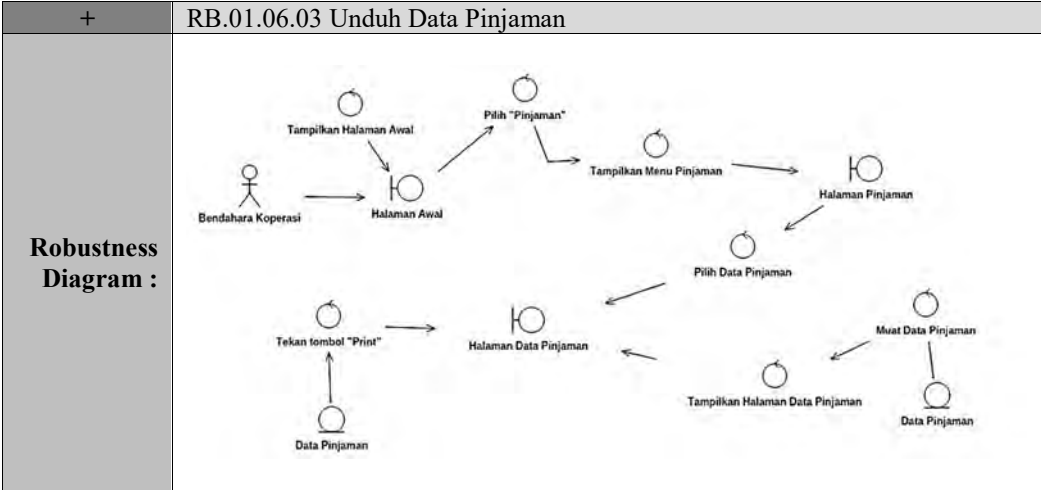
+	RB.01.06.02 Unduh Data Angsuran
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Angsuran dan Sistem menampilkan pilihan menu Angsuran</li> <li>4. Aktor memilih menu “Data Angsuran” dan Sistem menampilkan keseluruhan data Data Angsuran anggota</li> <li>5. Aktor menekan button “Print” pada baris data yang ingin diunduh, lalu Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor</li> </ol>
<b>Skenario Alternatif:</b>	-



Tabel B.22 RB Unduh Data Pinjaman

+	RB.01.06.03 Unduh Data Pinjaman
Deskripsi :	Robustness Diagram <b>Unduh Data Pinjaman</b> berdasarkan <i>Use Case Diagram Unduh Data Pinjaman</i>

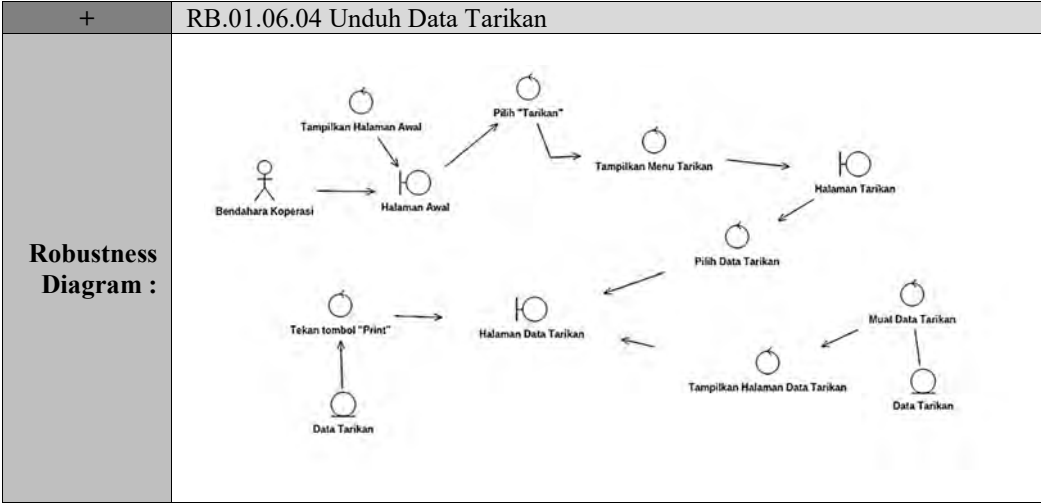
+	RB.01.06.03 Unduh Data Pinjaman
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Bendahara masuk ke halaman awal</li> <li>3. Aktor memilih menu Pinjaman dan Sistem menampilkan pilihan menu Pinjaman</li> <li>4. Aktor memilih menu “Data Pinjaman” dan Sistem menampilkan keseluruhan data Data Pinjaman anggota</li> <li>5. Aktor menekan button “Print” pada baris data yang ingin diunduh, lalu Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor</li> </ol>
<b>Skenario Alternatif:</b>	-



Tabel B.23 RB Unduh Data Tarikan

+	RB.01.06.04 Unduh Data Tarikan
Deskripsi :	Robustness Diagram Unduh Data Tarikan berdasarkan Use Case Diagram Unduh Data Tarikan

+	RB.01.06.04 Unduh Data Tarikan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Bendahara masuk ke halaman awal</li> <li>3. Aktor memilih menu Tarikan dan Sistem menampilkan pilihan menu Tarikan</li> <li>4. Aktor memilih menu “Data Tarikan” dan Sistem menampilkan keseluruhan data Data Tarikan anggota</li> <li>5. Aktor menekan button “Print” pada baris data yang ingin diunduh, lalu Sistem secara otomatis mengunduh data yang telah dipilih oleh aktor</li> </ol>
<b>Skenario Alternatif:</b>	<p style="text-align: center;">-</p>



Tabel B.24 RB Display Data Anggota

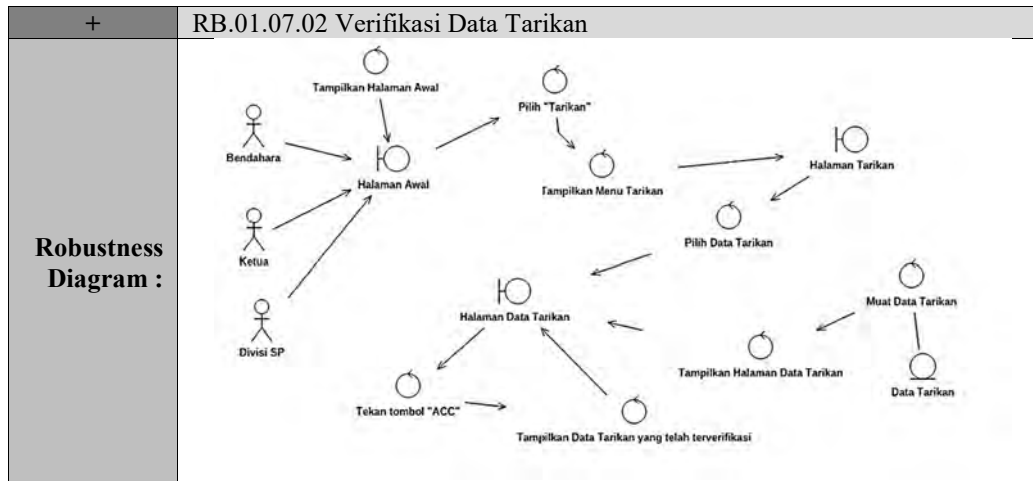
+	RB.01.07.01 Display Data Anggota
<b>Deskripsi :</b>	Robustness Diagram <b>Display Data Anggota</b> berdasarkan <i>Use Case Diagram Display Data Anggota</i>

+	RB.01.07.01 Display Data Anggota
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP, Ketua, dan Bendahara masuk ke halaman awal</li> <li>3. Aktor memilih menu Anggota dan Sistem menampilkan pilihan menu Anggota</li> <li>4. Aktor memilih menu “Data Anggota” dan Sistem menampilkan keseluruhan data Data Anggota anggota</li> </ol>
<b>Skenario Alternatif:</b>	-
<b>Robustness Diagram :</b>	<pre> graph TD     Bendahara --&gt; HA[Halaman Awal]     DivisiSP --&gt; HA     Ketua --&gt; HDA[Halaman Data Anggota]     THA[Tampilkan Halaman Awal] --&gt; HA     HA -- "Pilih 'Anggota'" --&gt; TMA[Tampilkan Menu Anggota]     TMA --&gt; HA_A[Halaman Anggota]     HA_A -- "Pilih Data Anggota" --&gt; DA[Data Anggota]     DA -- "Muat Data Anggota" --&gt; THDA[Tampilkan Halaman Data Anggota]     THDA --&gt; HDA   </pre>



**Tabel B.25 RB Verifikasi Data Tarikan**

+	RB.01.07.02 Verifikasi Data Tarikan
<b>Deskripsi :</b>	Robustness Diagram <b>Verifikasi Data Tarikan</b> berdasarkan <i>Use Case Diagram Verifikasi Data Tarikan</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP, Ketua, dan Bendahara masuk ke halaman awal</li> <li>3. Aktor memilih menu Tarikan dan Sistem menampilkan pilihan menu Tarikan</li> <li>4. Aktor memilih menu “Data Tarikan” dan Sistem menampilkan keseluruhan data Data Tarikan anggota</li> <li>5. Aktor menekan button “ACC” pada baris data yang ingin diubah, lalu Sistem menampilkan form dengan isian yang telah verifikasi</li> </ol>
<b>Skenario Alternatif:</b>	-



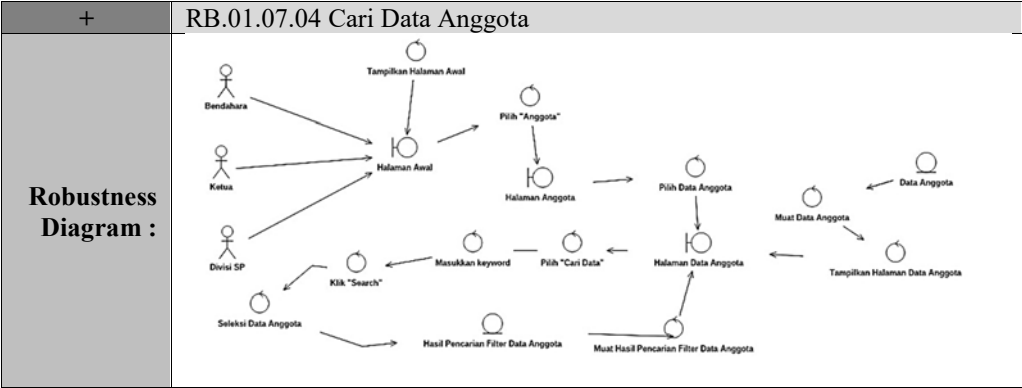
Tabel B.26 RB Verifikasi Data Pinjaman

+	RB.01.07.03 Verifikasi Data Pinjaman
<b>Deskripsi :</b>	Robustness Diagram <b>Verifikasi Data Pinjaman</b> berdasarkan <i>Use Case Diagram Verifikasi Data Pinjaman</i>

+	RB.01.07.03 Verifikasi Data Pinjaman
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP, Ketua, dan Bendahara masuk ke halaman awal</li> <li>3. Aktor memilih menu Pinjaman dan Sistem menampilkan pilihan menu Pinjaman</li> <li>4. Aktor memilih menu “Data Pinjaman” dan Sistem menampilkan keseluruhan data Data Pinjaman anggota</li> <li>5. Aktor menekan button “ACC” pada baris data yang ingin diubah, lalu Sistem menampilkan form dengan isian yang telah verifikasi</li> </ol>
<b>Skenario Alternatif:</b>	-
<b>Robustness Diagram :</b>	<pre> graph TD     Bendahara --&gt; HA[Halaman Awal]     Ketua --&gt; HA     DivisiSP --&gt; HA     HA --&gt; THA[Tampilkan Halaman Awal]     THA --&gt; PPin[Pilih "Pinjaman"]     PPin --&gt; TMP[Tampilkan Menu Pinjaman]     TMP --&gt; HP[Halaman Pinjaman]     HP --&gt; PDP[Pilih Data Pinjaman]     PDP --&gt; HDP[Halaman Data Pinjaman]     HDP --&gt; MDP[Muat Data Pinjaman]     MDP --&gt; DP[Data Pinjaman]     DP --&gt; THDP[Tampilkan Halaman Data Pinjaman]     THDP --&gt; TACC[Tekan tombol "ACC"]     TACC --&gt; TDPV[Tampilkan Data Pinjaman yang telah terverifikasi]   </pre>

**Tabel B.27 RB Cari Data Anggota**

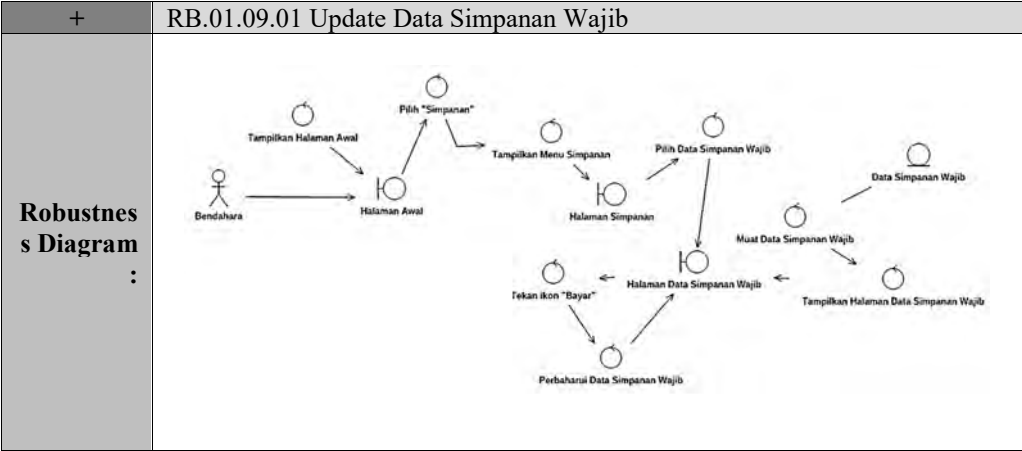
+	RB.01.07.04 Cari Data Anggota
<b>Deskripsi :</b>	Robustness Diagram <b>Verifikasi Data Anggota</b> berdasarkan <i>Use Case Diagram Verifikasi Data Anggota</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP, Ketua, dan Bendahara masuk ke halaman awal</li> <li>3. Aktor memilih menu Anggota dan Sistem menampilkan pilihan menu Anggota</li> <li>4. Aktor memilih menu “Data Anggota” dan Sistem menampilkan keseluruhan data Data Anggota anggota</li> <li>5. Aktor memilih atribut filter data, dan memasukkan keyword yang ingin dicari</li> <li>6. Sistem menampilkan rekap data sesuai dengan pilihan filter data</li> </ol>
<b>Skenario Alternatif:</b>	-



**Tabel B.28 RB Update Data Simpanan Wajib**

+	RB.01.09.01 Update Data Simpanan Wajib
<b>Deskripsi :</b>	Robustness Diagram <b>Update Data Simpanan Wajib</b> berdasarkan <i>Use Case Diagram Update Data Simpanan Wajib</i>

+	RB.01.09.01 Update Data Simpanan Wajib
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor memilih menu Simpanan Sistem menampilkan pilihan menu Simpanan</li> <li>2. Aktor memilih menu “Data Simpanan Wajib” Sistem menampilkan keseluruhan data Simpanan Wajib</li> <li>3. Aktor memasukkan “Pilih Bulan” dan “Pilih Tahun”</li> <li>4. Aktor menekan tombol “Lihat” Sistem menampilkan data yang telah di filter oleh aktor</li> <li>5. Aktor menekan icon “Bayar” apabila anggota telah membayar Sistem menampilkan data dengan isian yang telah diubah dalam bentuk tabel</li> </ol>
<b>Skenario Alternatif:</b>	-



**Tabel B.29 Tambah Data Simpanan**

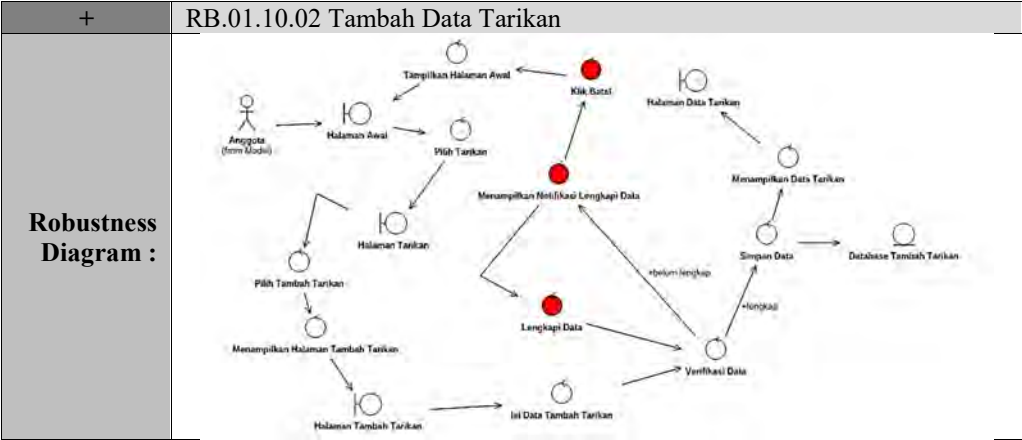
+	RB.01.10.03 Tambah Data Simpanan
<b>Deskripsi :</b>	Robustness Diagram <b>Tambah Data Simpanan</b> berdasarkan <i>Use Case Diagram Tambah Data Simpanan</i>

+	RB.01.10.03 Tambah Data Simpanan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Anggota masuk ke halaman awal</li> <li>3. Aktor memilih menu Pinjaman dan Sistem menampilkan pilihan menu Pinjaman</li> <li>4. Aktor memilih menu “Tambah Pinjaman” dan Sistem menampilkan form Tambah Pinjaman</li> <li>5. Aktor mengisi data tambah Pinjaman, lalu Sistem menampilkan form tambah data Pinjaman</li> <li>6. Aktor menekan tombol Simpan, lalu Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem akan menampilkan pemberitahuan “Anda harus membayarkan minimal 50 % dari pinjaman Anda sebelumnya!”</li> <li>2. Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem tidak menampilkan tombol simpan</li> <li>3. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”</li> <li>4. Jika aktor menekan tombol reset maka sistem akan otomatis menghapus semua data yang telah diisikan didalam form</li> </ol>





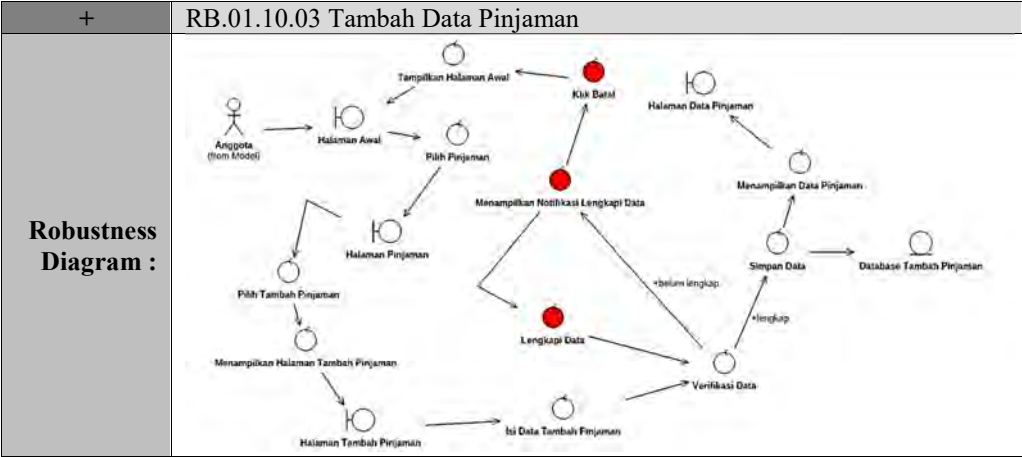
+	RB.01.10.02 Tambah Data Tarikan
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Anggota masuk ke halaman awal</li> <li>3. Aktor memilih menu Tarikan dan Sistem menampilkan pilihan menu Tarikan</li> <li>4. Aktor memilih menu “Tambah Tarikan” dan Sistem menampilkan form Tambah Tarikan</li> <li>5. Aktor mengisi data tambah tarikan, lalu Sistem menampilkan form tambah data tarikan</li> <li>6. Aktor menekan tombol Submit, lalu Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”</li> <li>2. Jika aktor menekan tombol batal maka sistem akan menampilkan pemberitahuan “Apakah anda yakin untuk membatalkan transaksi?” lalu kembali ke halaman data tarikan</li> </ol>



Tabel B.31 RB Tambah Data Pinjaman

+	RB.01.10.03 Tambah Data Pinjaman
Deskripsi :	Robustness Diagram Tambah Data Pinjaman berdasarkan Use Case Diagram Tambah Data Pinjaman

+	RB.01.10.03 Tambah Data Pinjaman
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. Anggota masuk ke halaman awal</li> <li>3. Aktor memilih menu Pinjaman dan Sistem menampilkan pilihan menu Pinjaman</li> <li>4. Aktor memilih menu “Tambah Pinjaman” dan Sistem menampilkan form Tambah Pinjaman</li> <li>5. Aktor mengisi data tambah Pinjaman, lalu Sistem menampilkan form tambah data Pinjaman</li> <li>6. Aktor menekan tombol Simpan, lalu Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem akan menampilkan pemberitahuan “Anda harus membayarkan minimal 50 % dari pinjaman Anda sebelumnya!”</li> <li>2. Jika anggota belum membayar minimal 50% dari pinjaman sebelumnya, maka sistem tidak menampilkan tombol simpan</li> <li>3. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”</li> <li>4. Jika aktor menekan tombol reset maka sistem akan otomatis menghapus semua data yang telah diisikan didalam form</li> </ol>



Tabel B.32 RB Tambah Data Anggota

+	RB.01.11.01 Tambah Data Anggota
<b>Deskripsi :</b>	Robustness Diagram <b>Tambah Data Anggota</b> berdasarkan <i>Use Case Diagram Tambah Data Anggota</i>

+	RB.01.11.01 Tambah Data Anggota
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Anggota dan Sistem menampilkan pilihan menu Anggota</li> <li>4. Aktor memilih menu “Tambah Anggota” dan Sistem menampilkan form Tambah Anggota</li> <li>5. Aktor mengisi data tambah Anggota, lalu Sistem menampilkan form tambah data Anggota</li> <li>6. Aktor menekan tombol Submit, lalu Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</li> </ol>

+	RB.01.11.01 Tambah Data Anggota
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika terdapat field pada formulir yang masih kosong maka sistem menampilkan pesan “Harap isi bidang ini” dan terdapat notifikasi “tidak boleh kosong” pada bagian kanan form</li> <li>2. Jika aktor tidak memasukkan tanggal secara keseluruhan, maka sistem akan menampilkan pemberitahuan “Masukkan nilai yang valid. Bidang tersebut tidak lengkap atau memiliki tanggal yang tidak valid”.</li> <li>3. Nomor telepon dan gaji yang dimasukkan harus berupa angka</li> <li>4. Jika aktor menekan tombol batal maka sistem akan menampilkan halaman awal</li> </ol>
<b>Robustness Diagram :</b>	

**Tabel B.33 RB Tambah Daftar Tagihan Angsuran**

+	RB.01.11.02 Tambah Daftar Tagihan Angsuran
<b>Deskripsi :</b>	Robustness Diagram <b>Tambah Daftar Tagihan Angsuran</b> berdasarkan <i>Use Case Diagram Tambah Daftar Tagihan Angsuran</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"> <li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li> <li>2. DivisiSP masuk ke halaman awal</li> <li>3. Aktor memilih menu Angsuran dan Sistem menampilkan pilihan menu Angsuran</li> <li>4. Aktor memilih “Tambah Daftar Tagihan Angsuran” dan Sistem menampilkan form Tambah Angsuran</li> <li>5. Aktor mengisi data tambah Daftar Tagihan Angsuran, lalu Sistem menampilkan form tambah daftar tagihan Angsuran</li> <li>6. Aktor menekan tombol Submit, lalu Sistem menampilkan notifikasi sukses dan menyimpan data kedalam database</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Jika field “Jumlah Bayar” pada formulir yang masih kosong maka sistem menampilkan pesan “Lengkapi Data!”</li> <li>2. Jika aktor belum memilih “ID Pinjaman Belum Lunas” maka sistem menampilkan pesan “Lengkapi Data!”</li> </ol>





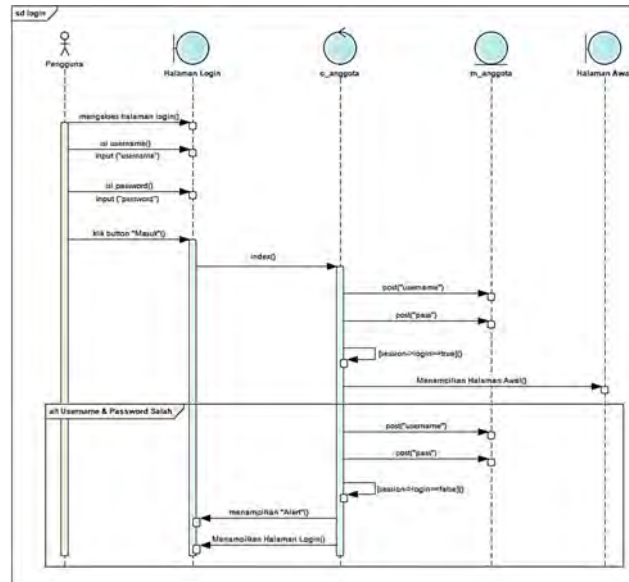
Tabel B.34 RB Edit Akun

+	RB.01.12.01 Edit Akun
<b>Deskripsi :</b>	Robustness Diagram <b>Edit Akun</b> berdasarkan <i>Use Case Diagram Edit Akun</i>
<b>Skenario Sukses Utama :</b>	<ol style="list-style-type: none"><li>1. Aktor membuka halaman awal aplikasi, dan Sistem menampilkan halaman home</li><li>2. DivisiSP masuk ke halaman awal</li><li>3. Pengguna memilih fitur “Pengaturan Akun” Sistem menampilkan halaman pengaturan akun</li><li>4. Pengguna merubah Nama dan Password lama dengan yang baru</li></ol>

+	RB.01.12.01 Edit Akun
	<ol style="list-style-type: none"> <li>5. Pengguna menekan button “Simpan”</li> <li>6. Sistem menampilkan halaman awal</li> </ol>
<b>Skenario Alternatif:</b>	<ol style="list-style-type: none"> <li>1. Pengguna batal merubah data nama dan password dengan menekan tombol “Batal”. Sistem menampilkan alert “Apakah anda yakin untuk batal mengubah data anggota?”</li> <li>2. Jika password yang diinputkan pada “Ulangi Password Baru” tidak sesuai dengan “Password Baru” yang telah dimasukkan, maka akan muncul pemberitahuan “Password yang anda masukkan tidak sesuai!”</li> </ol>
<b>Robustness Diagram :</b>	<pre> graph TD     U1((Masukkan Username &amp; Password)) --&gt; U2((Tampilkan Halaman Login))     U2 --&gt; U3((Halaman Login))     U3 --&gt; U4((Tampilkan Halaman Awal))     U4 --&gt; U5((Halaman Awal))     U5 --&gt; U6((Menampilkan Pengaturan Akun))     U6 --&gt; U7((Pilih Pengaturan Akun))     U7 --&gt; U8((Klik button "Simpan"))     U8 --&gt; U9((Ubah Nama &amp; Password))     U9 --&gt; U10((Halaman Pengaturan Akun))     U10 --&gt; U11((Menampilkan Halaman Pengaturan Akun))     U11 -- "+data lengkap" --&gt; U5     U11 -- "+data belum lengkap" --&gt; U11     U11 --&gt; U12((Menampilkan Alert Lengkapi Data))     U12 --&gt; U5     U12 --&gt; U11   </pre>

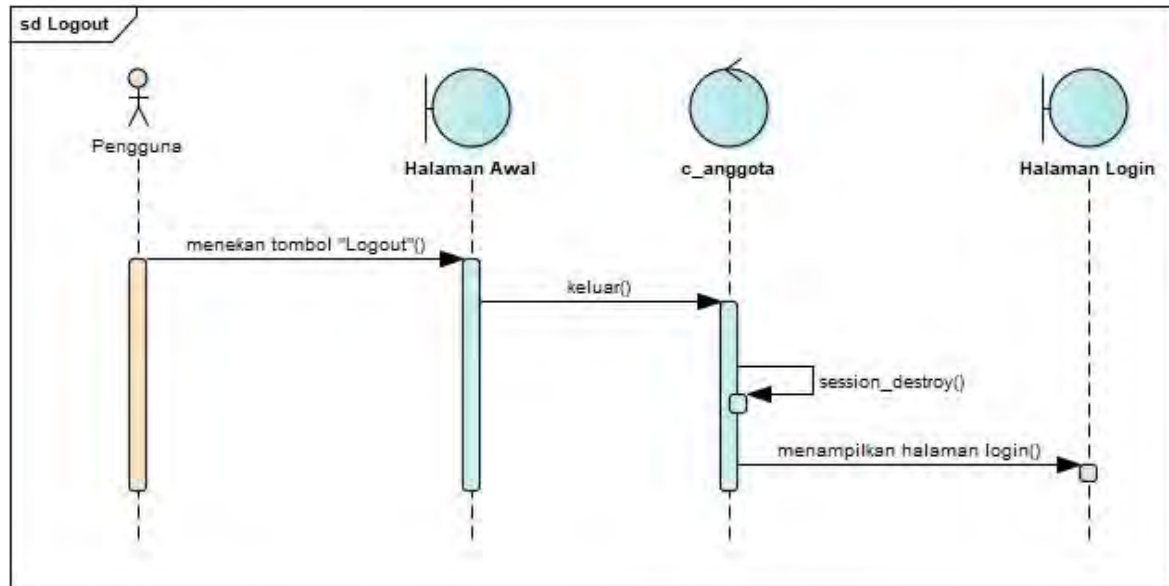
## B.2 Sequence Diagram Sistem Informasi Koperasi PT Wonosari Jaya Pengguna

- Login



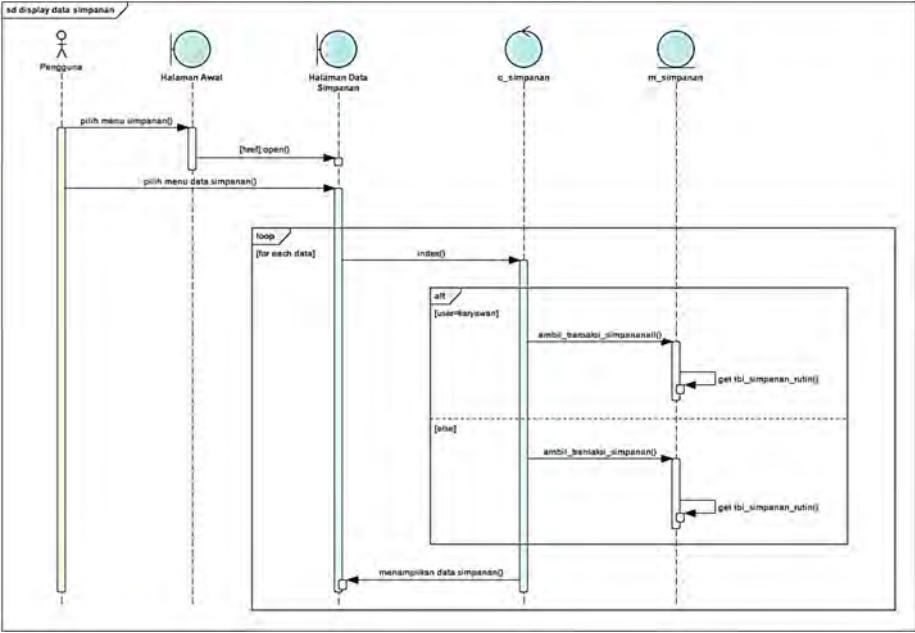
Gambar B.1 Sequence Login

- Logout



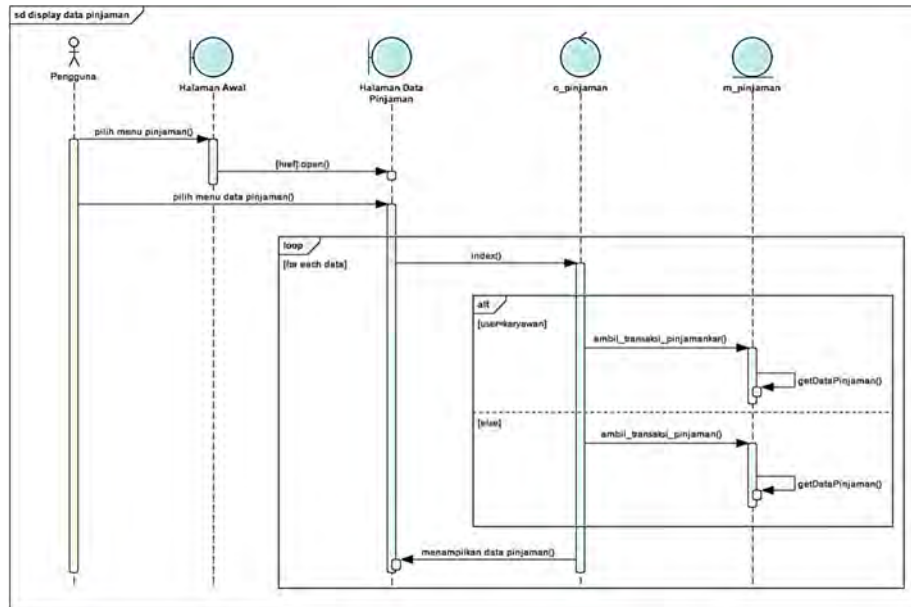
Gambar B.2 Sequence Logout

- Display Data Simpanan



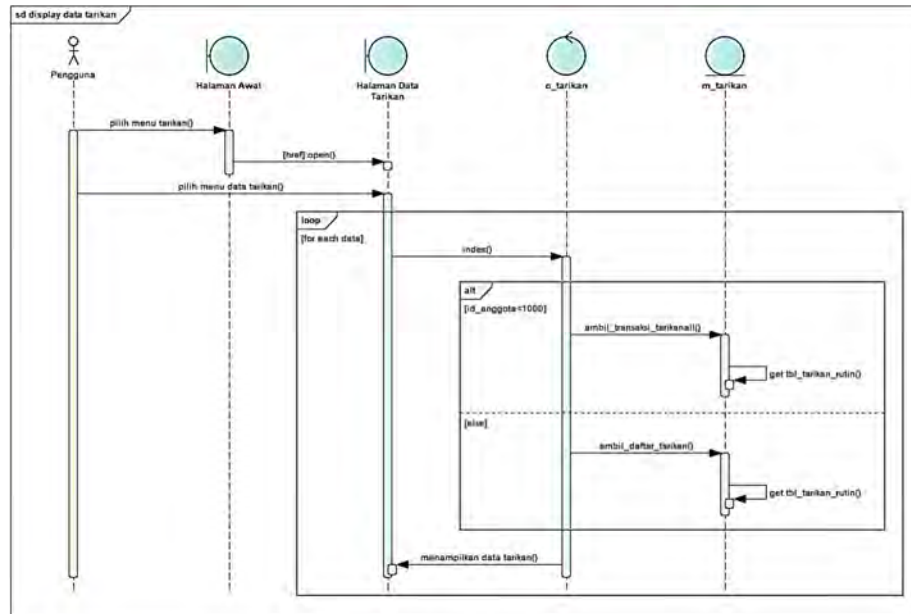
Gambar B.3 Display Data Simpanan

- Display Data Pinjaman



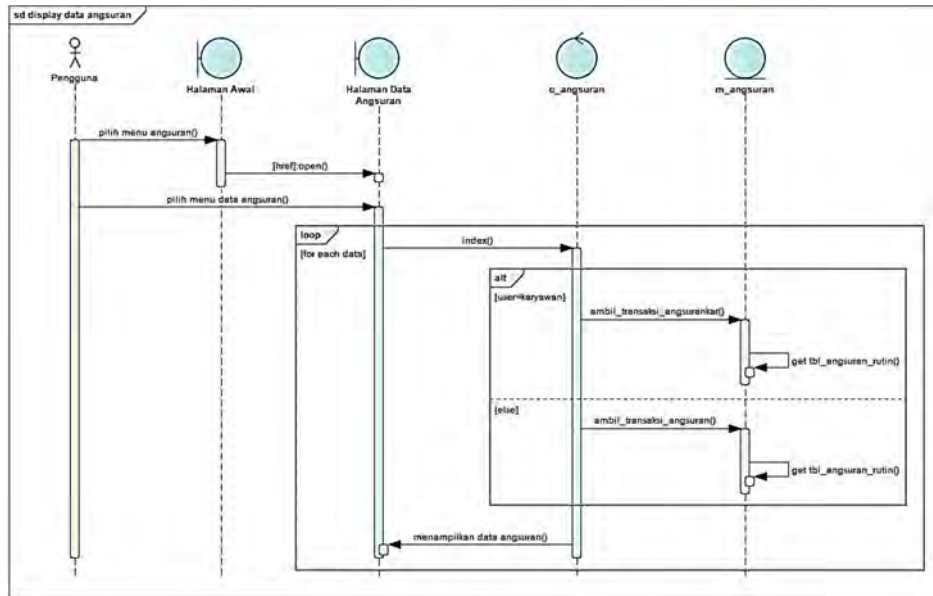
Gambar B.4 Display Data Pinjaman

- Display Data Tarikan



Gambar B.5 Display Data Tarikan

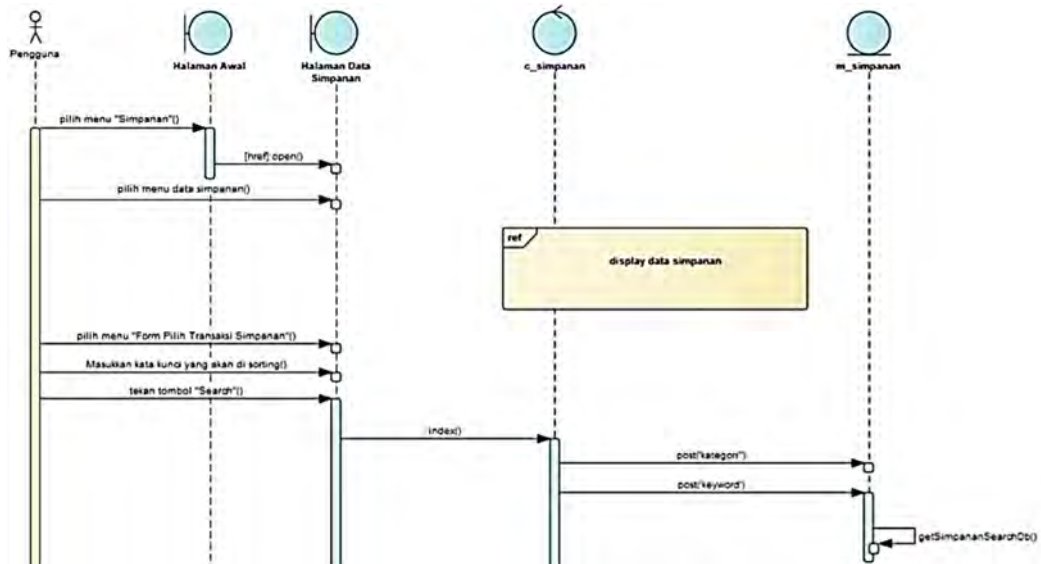
- Display Data Angsuran

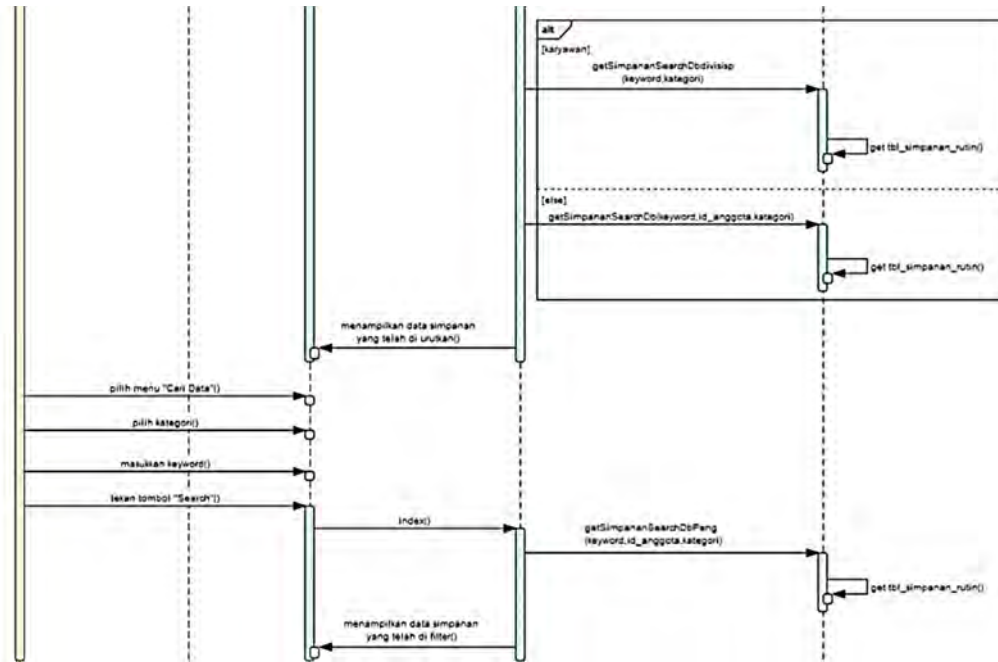


Gambar B.6 Display Data Angsuran



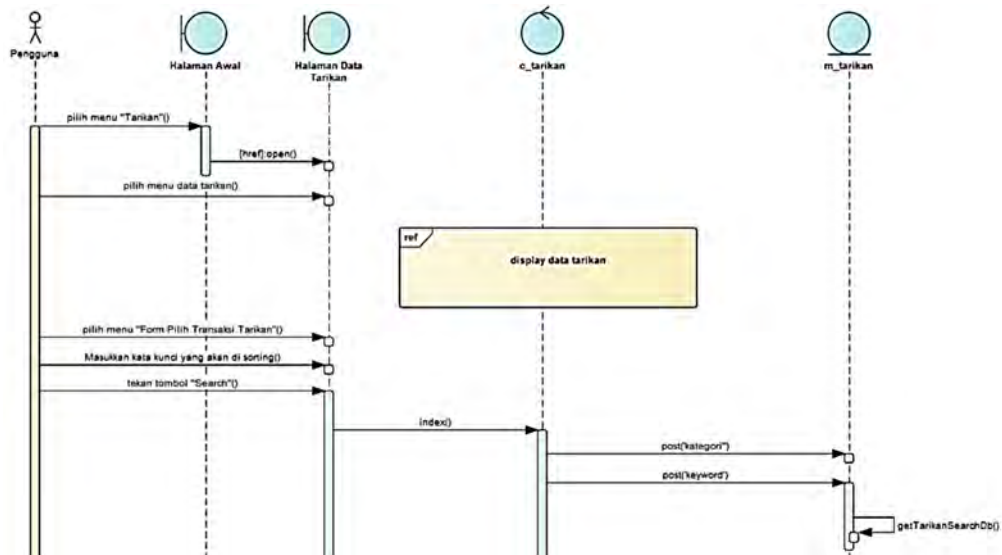
- Cari Data Simpanan

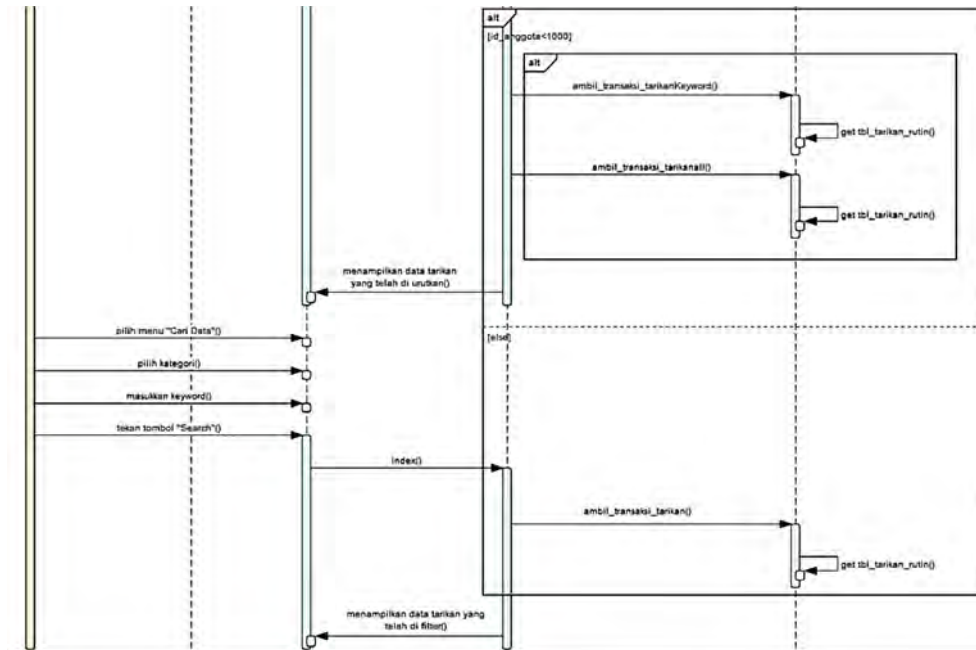




Gambar B.7 Cari Data Simpanan

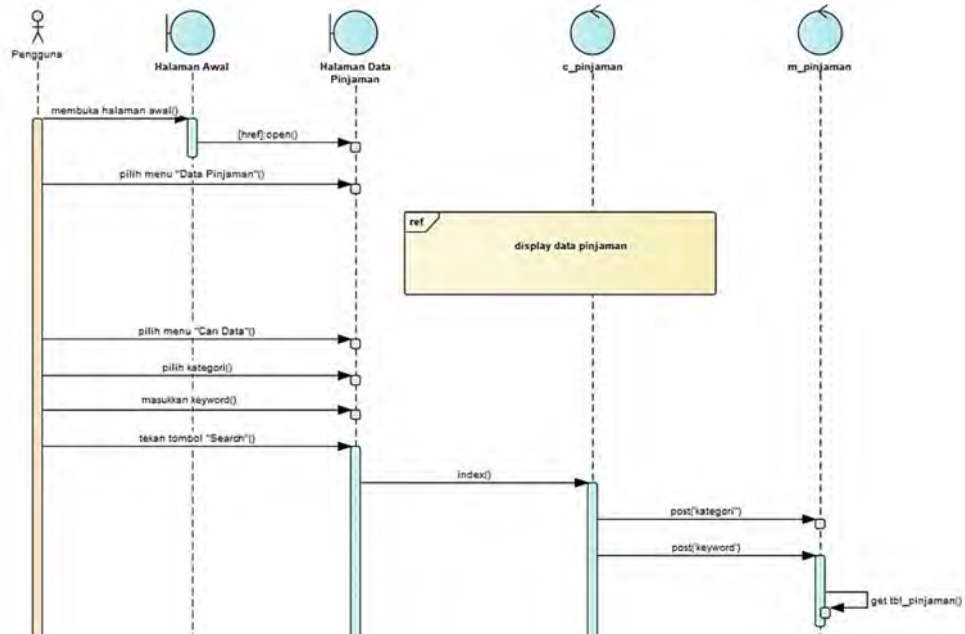
- Cari Data Tarikan

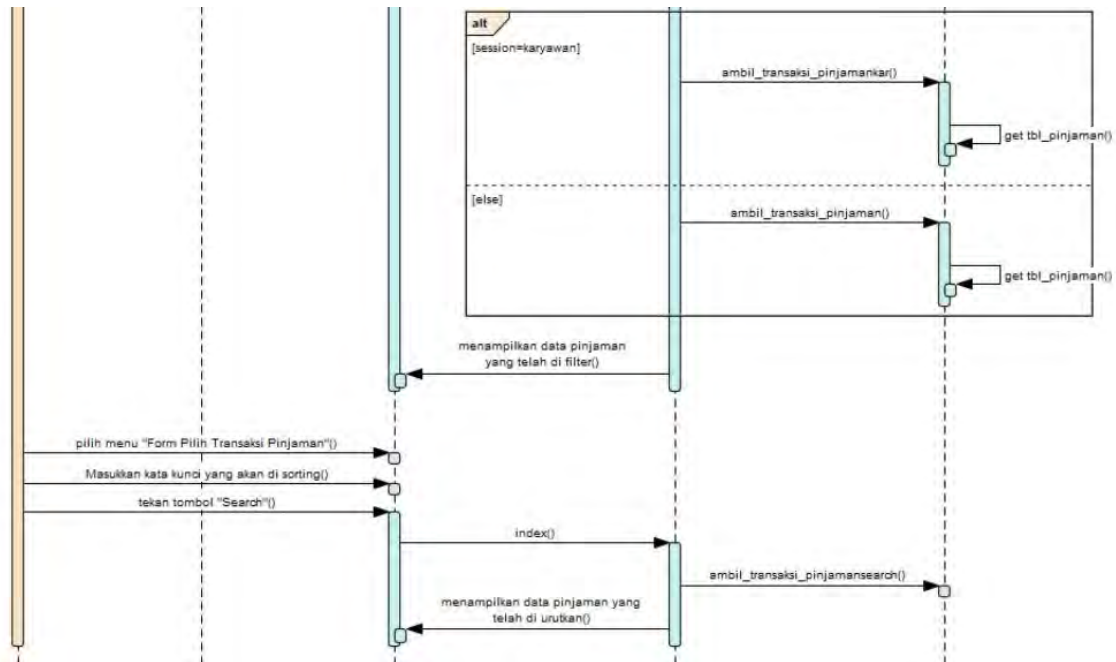




Gambar B.8 Cari Data Tarikan

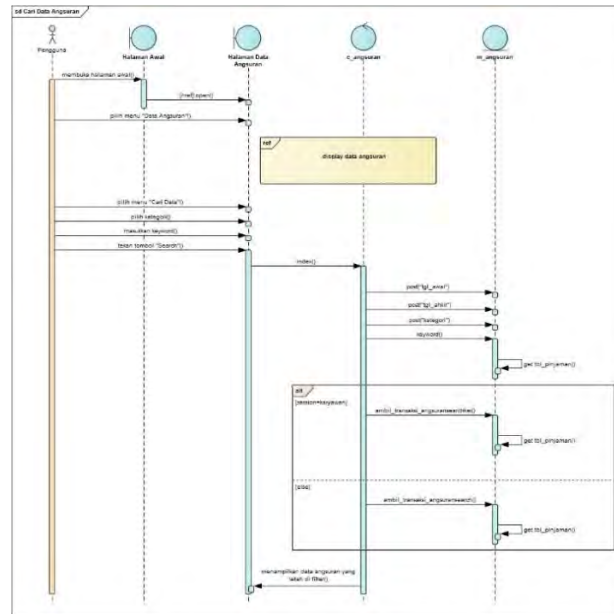
- Cari Data Pinjaman





Gambar B.9 Cari Data Pinjaman

- Cari Data Angsuran

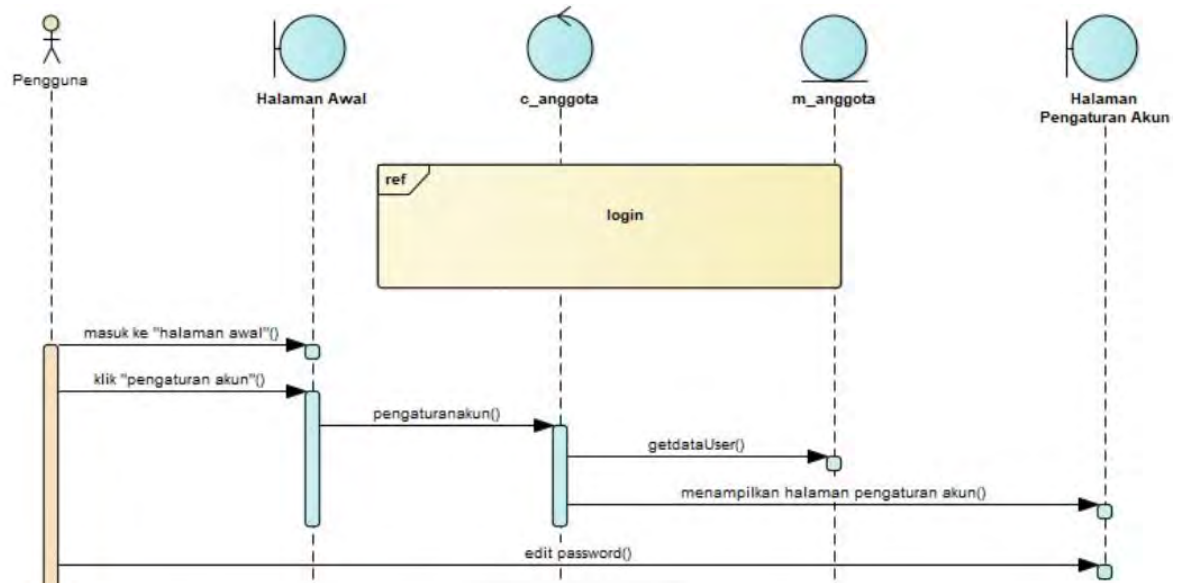


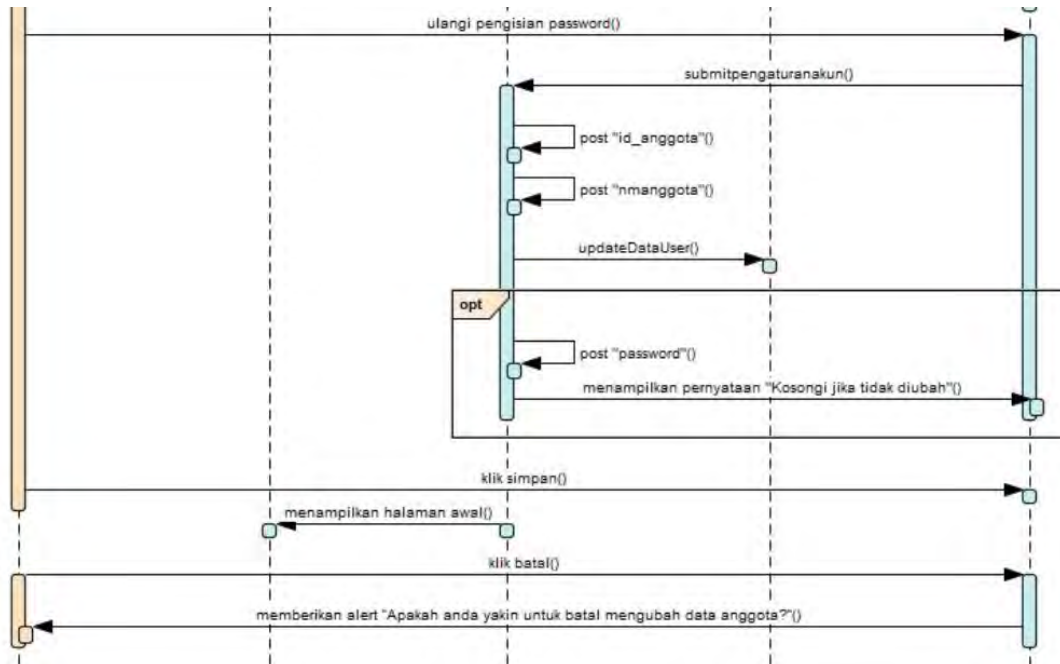
### Gambar B.10 Cari Data Angsuran

*“Halaman ini sengaja dikosongkan”*



- Edit Akun

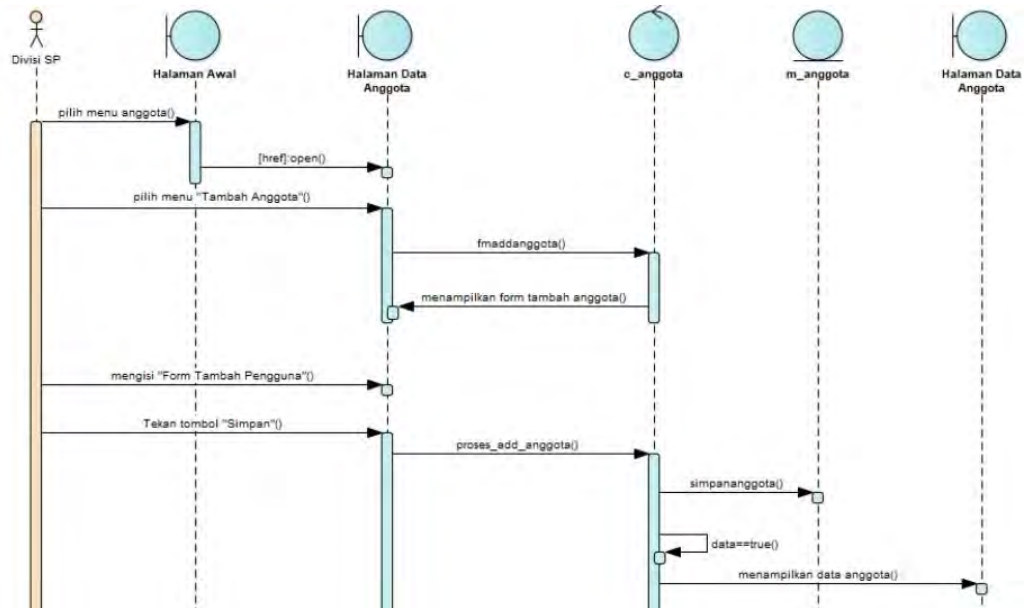


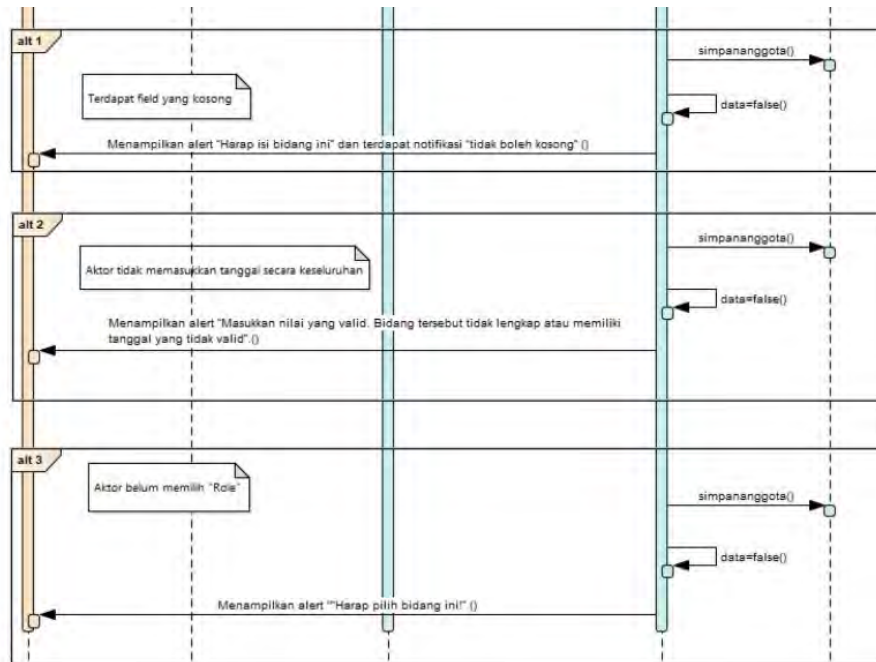


Gambar B.11 Edit Akun

## Divisi SP

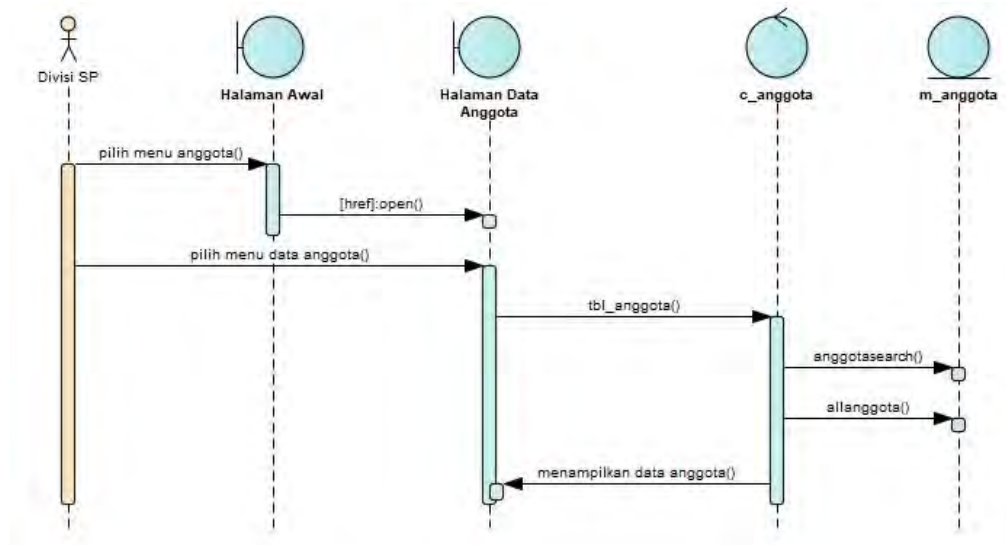
- Tambah Data Anggota





Gambar B.12 Tambah Data Anggota

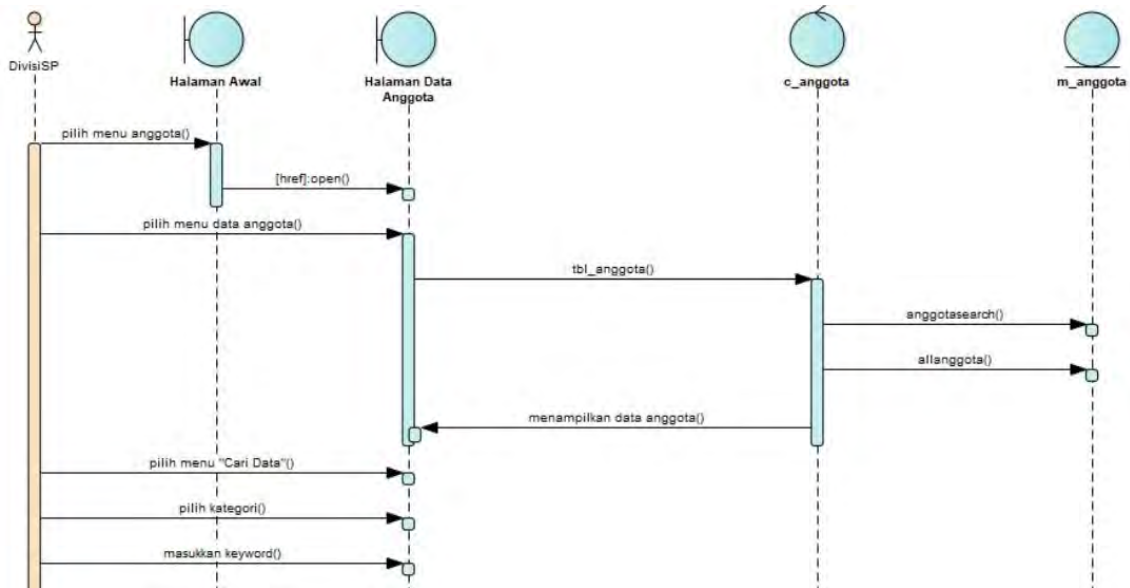
- Display Data Anggota

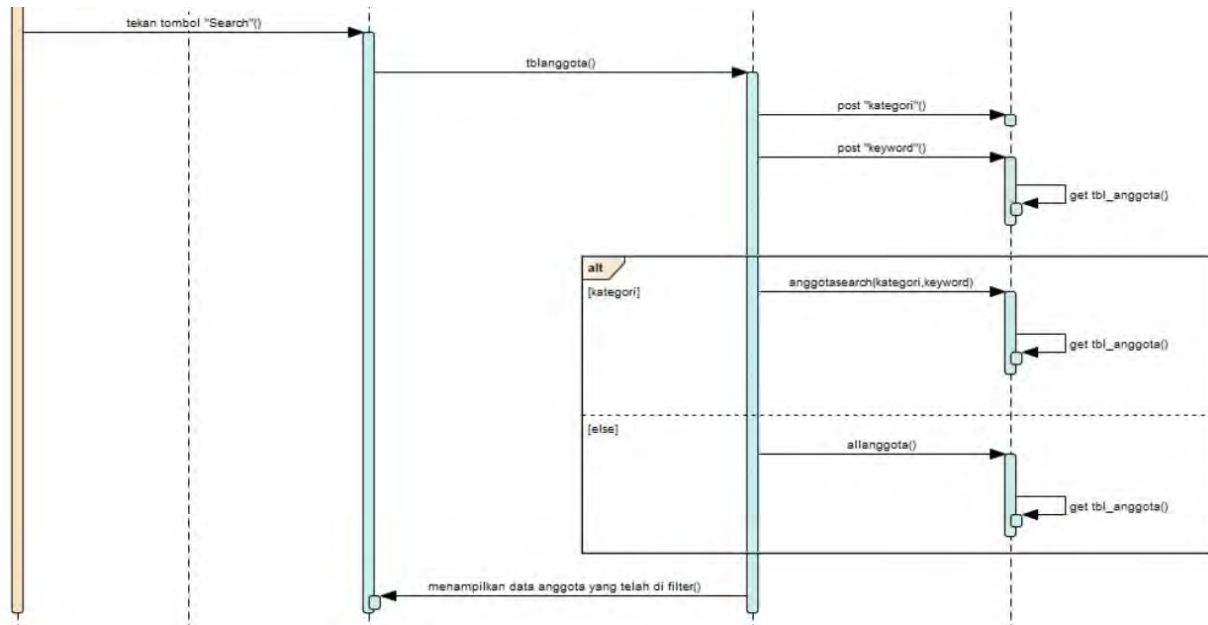


Gambar B.13 Display Data Anggota

*“Halaman ini sengaja dikosongkan”*

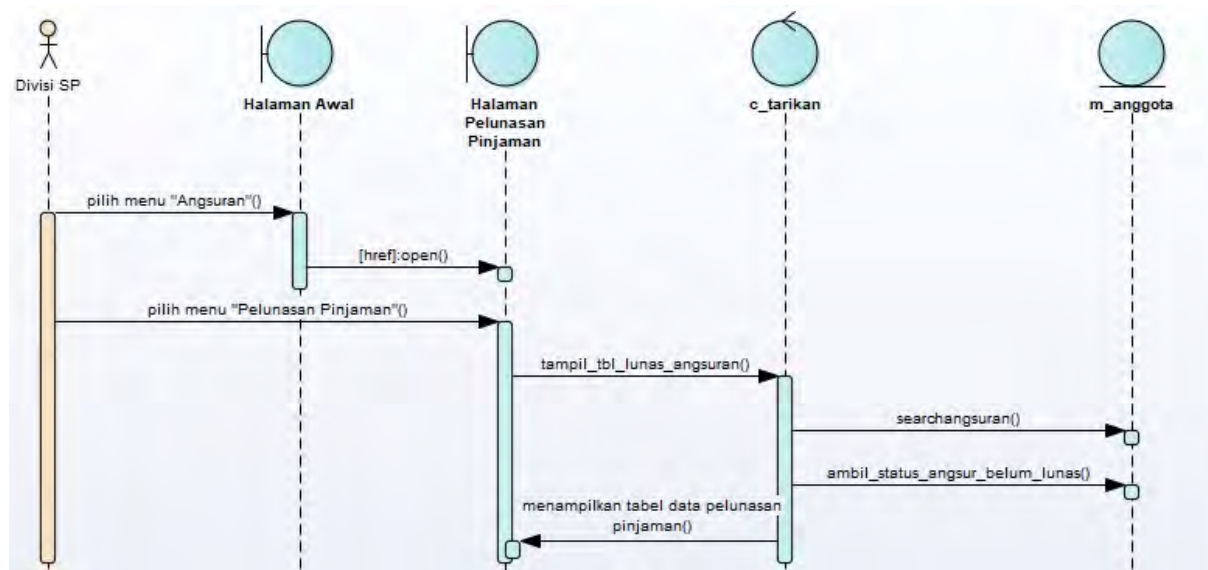
- Cari Data Anggota



**Gambar B.14 Cari Data Anggota**

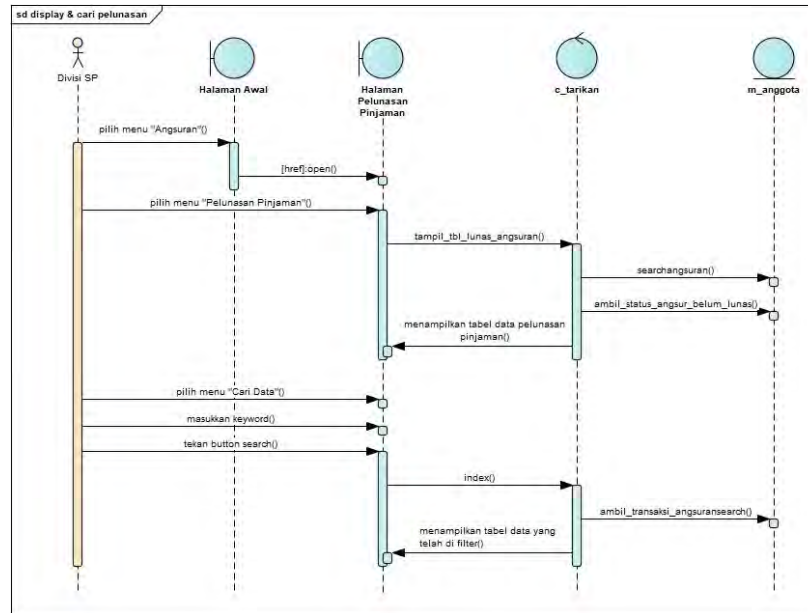


- Display Data Pelunasan Pinjaman



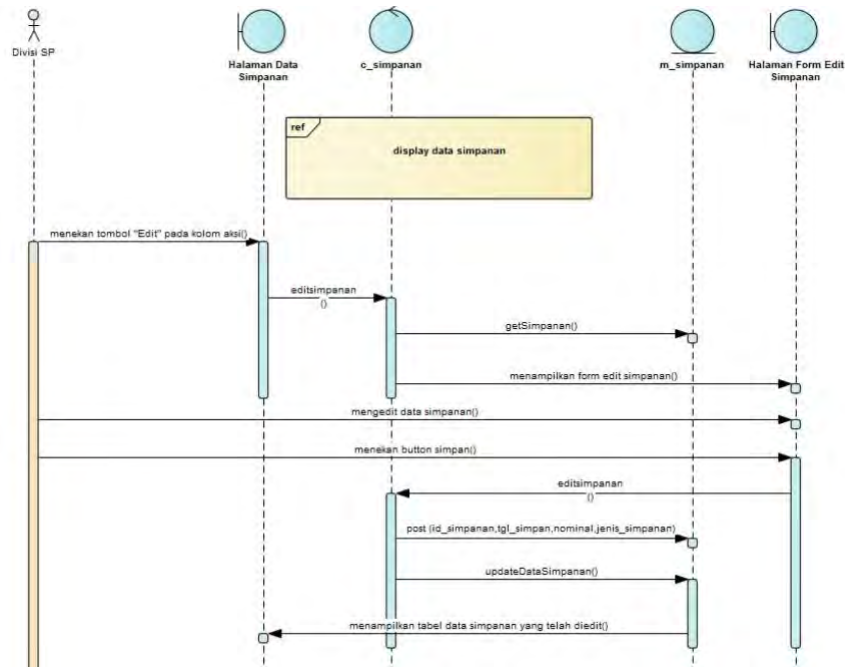
Gambar B.15 Display Data Pelunasan Pinjaman

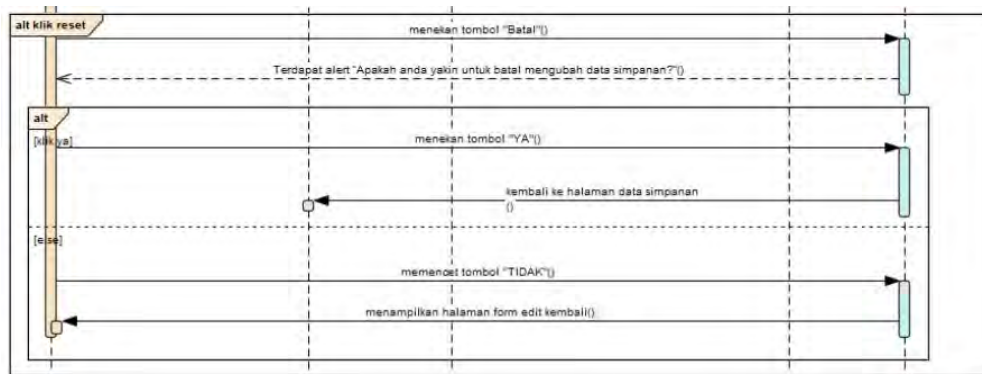
- Cari Data Pelunasan Pinjaman



Gambar B.16 Cari Data Pelunasan Pinjaman

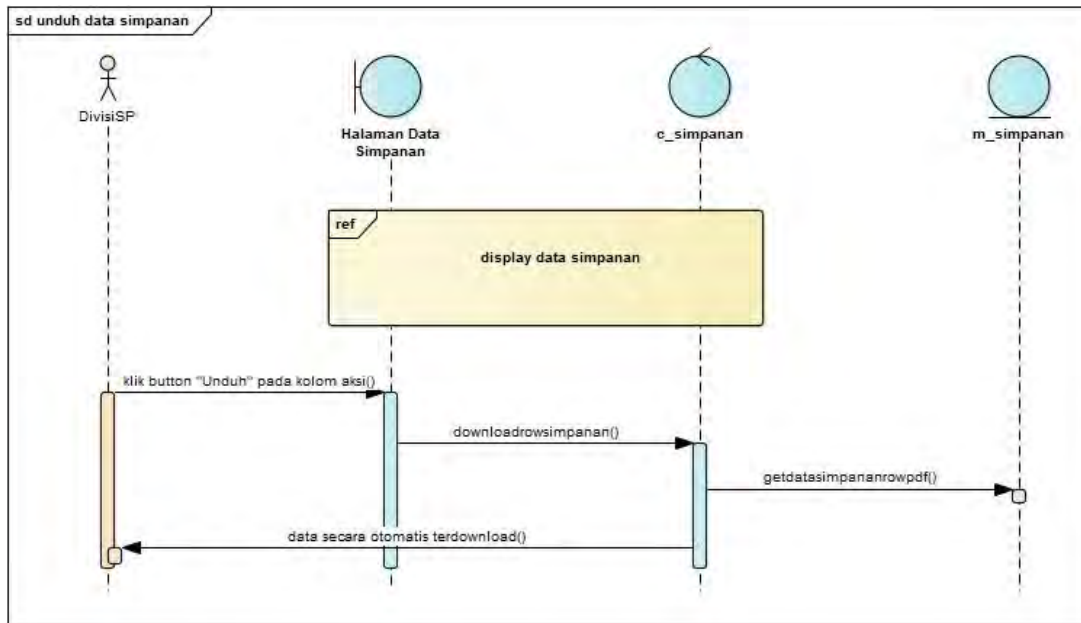
- Edit Data Simpanan





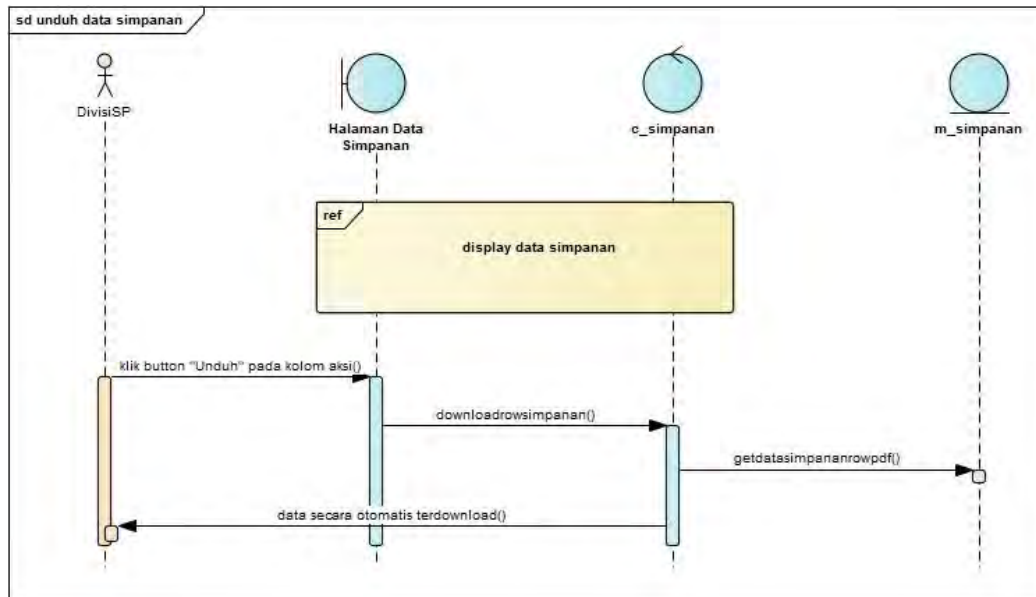
**Gambar B.17 Edit Data Simpanan**

- Unduh Data Simpanan



Gambar B.18 Unduh Data Simpanan

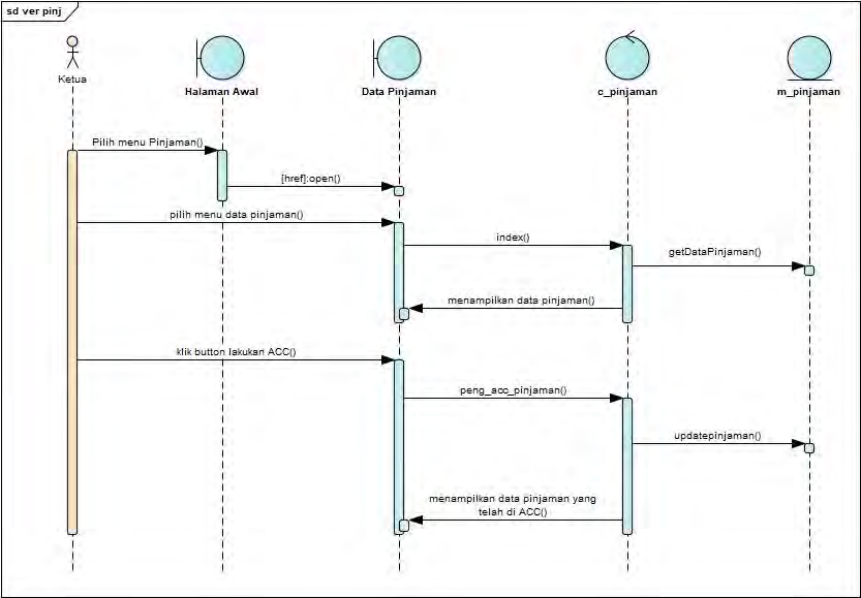
➤ Unduh Data Angsuran



Gambar B.19 Unduh Data Angsuran

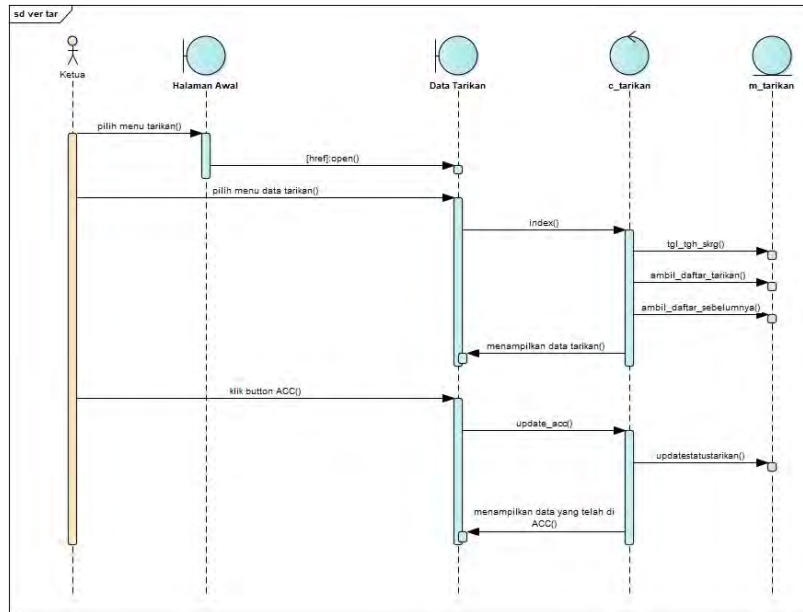
# Ketua Koperasi

- Verifikasi Data Pinjaman



Gambar B.20 Verifikasi Data Pinjaman

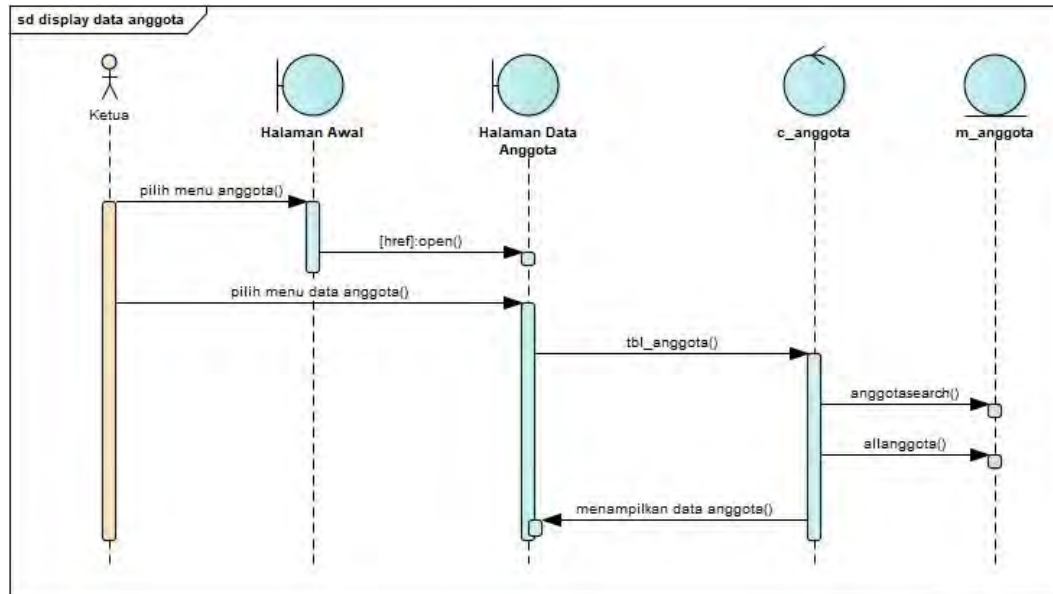
- Verifikasi Data Tarikan



**Gambar B.21 Verifikasi Data Tarikan**



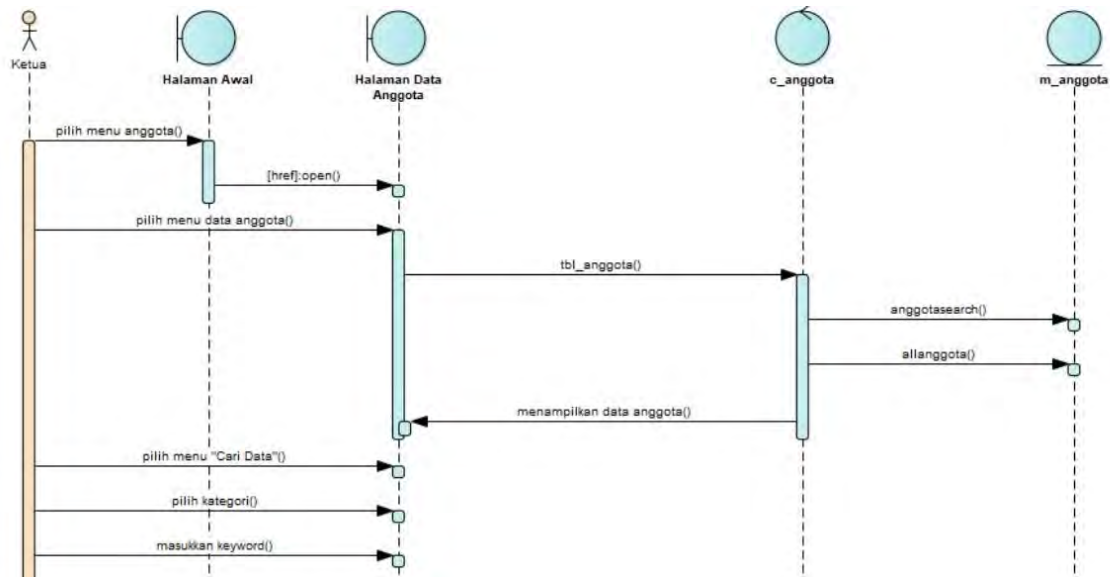
- Display Data Anggota

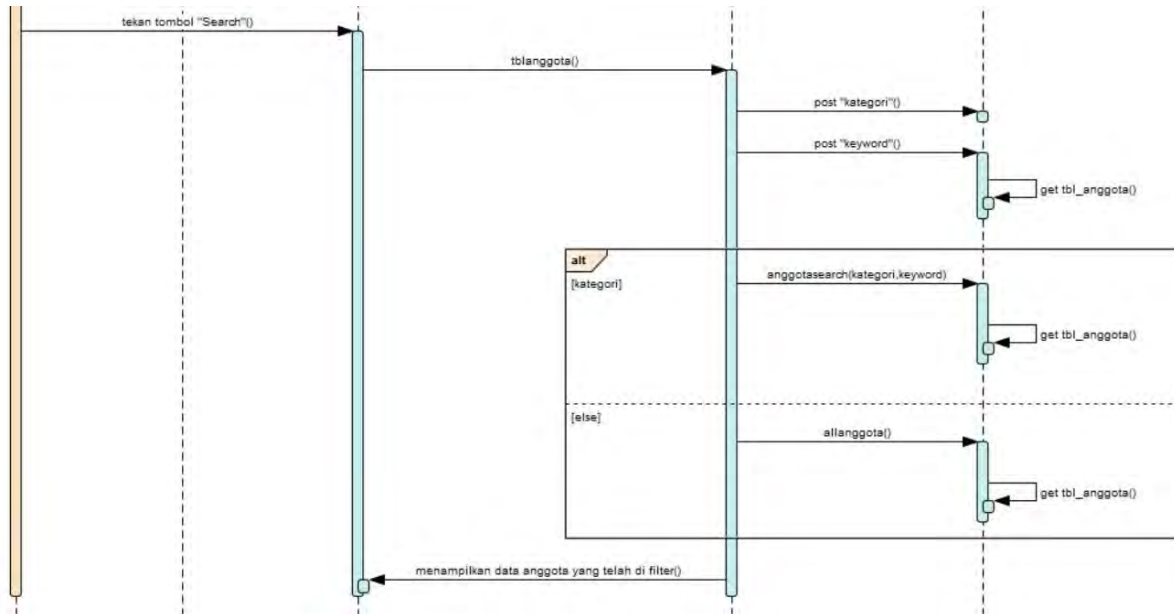


Gambar B.22 Display Data Anggota

*“Halaman ini sengaja dikosongkan”*

- Cari Data Anggota

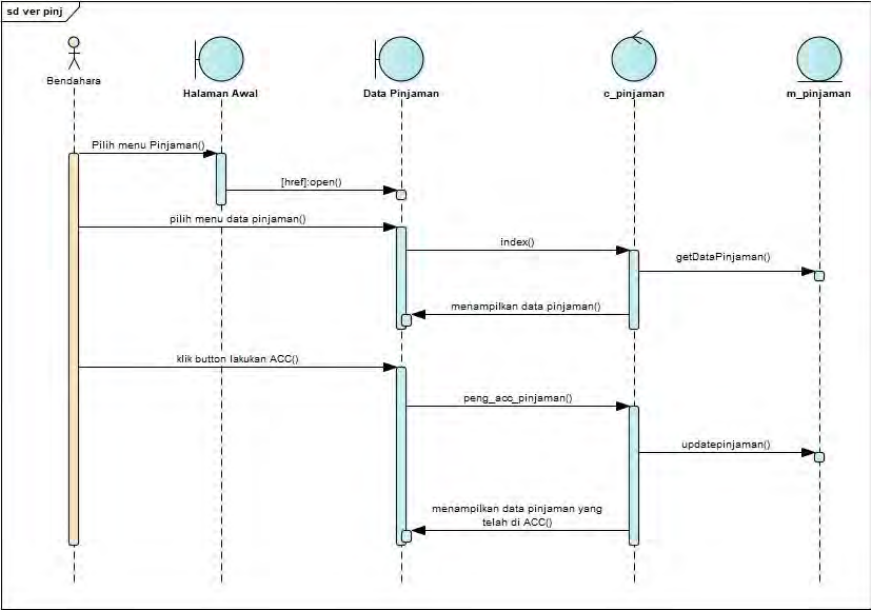




Gambar B.23 Cari Data Anggota

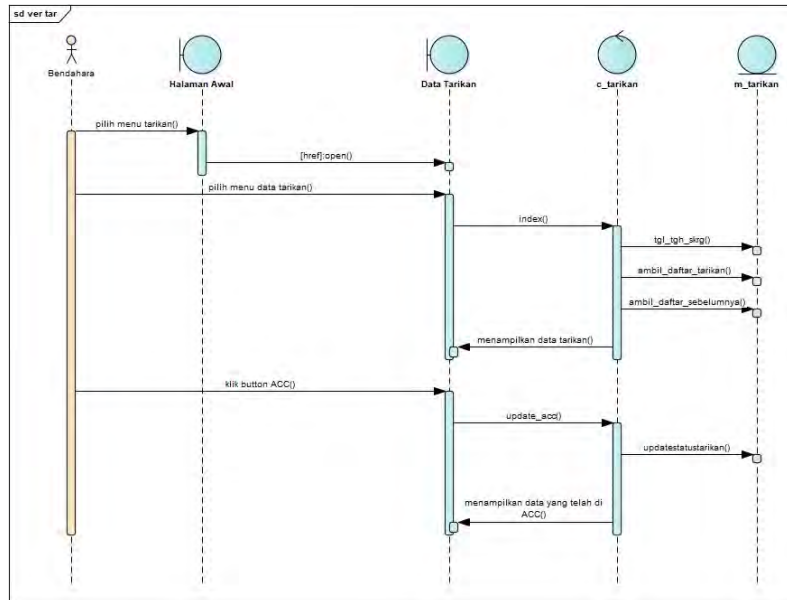
# Bendahara Koperasi

- Verifikasi Data Pinjaman



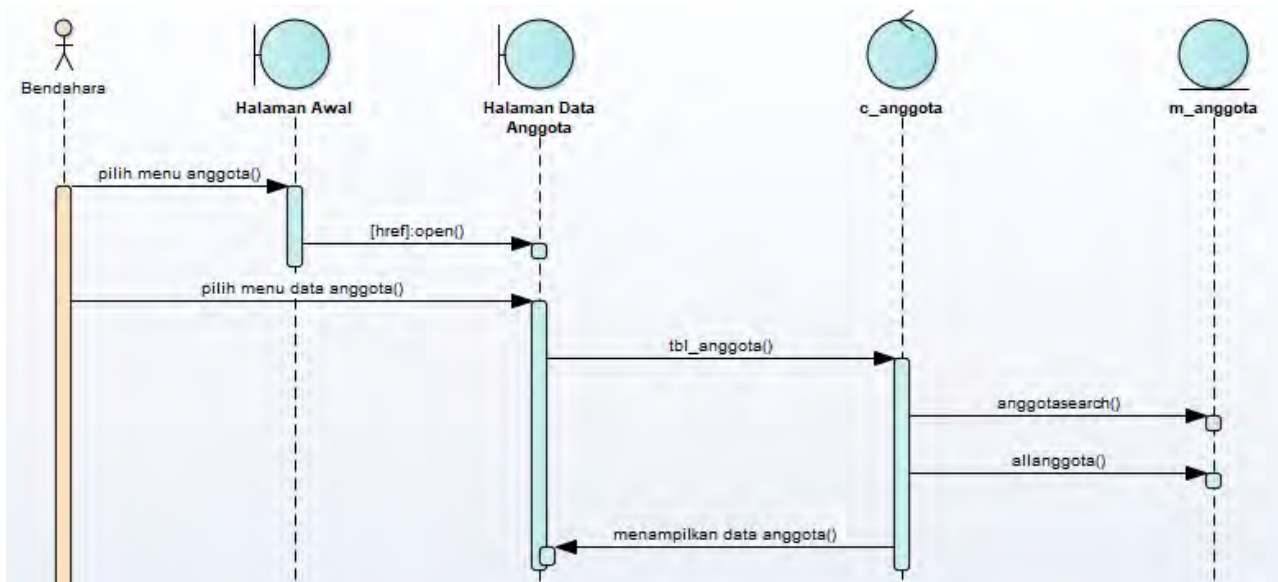
Gambar B.24 Verifikasi Data Pinjaman

- Verifikasi Data Tarikan



Gambar B.25 Verifikasi Data Tarikan

- Display Data Anggota

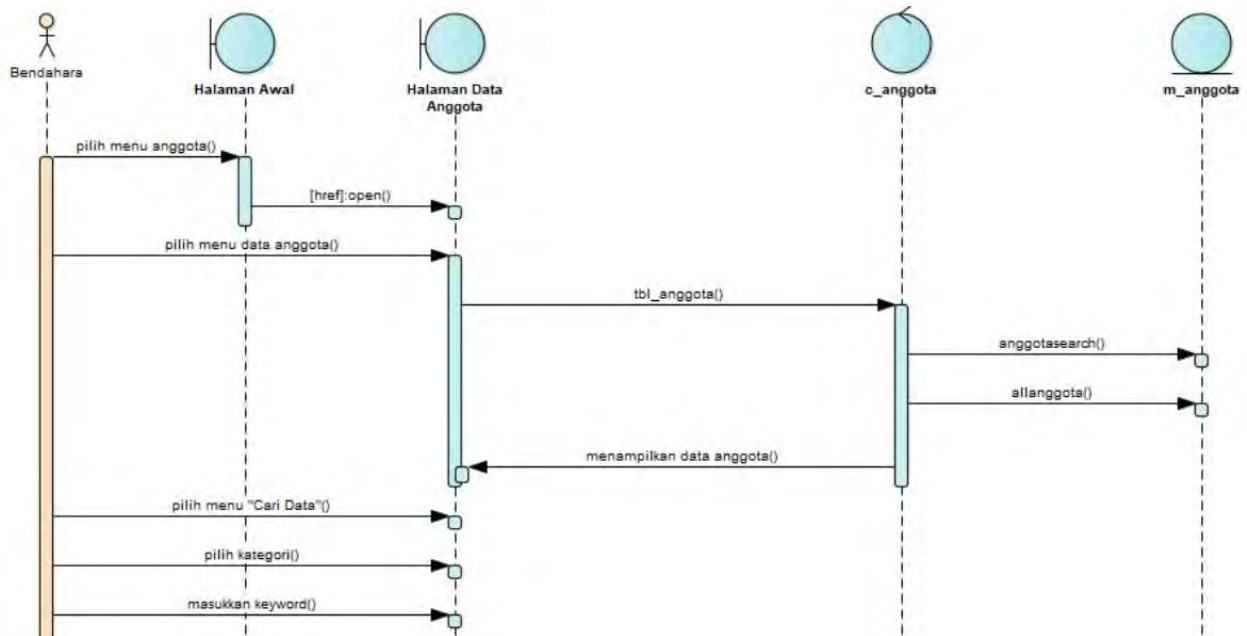


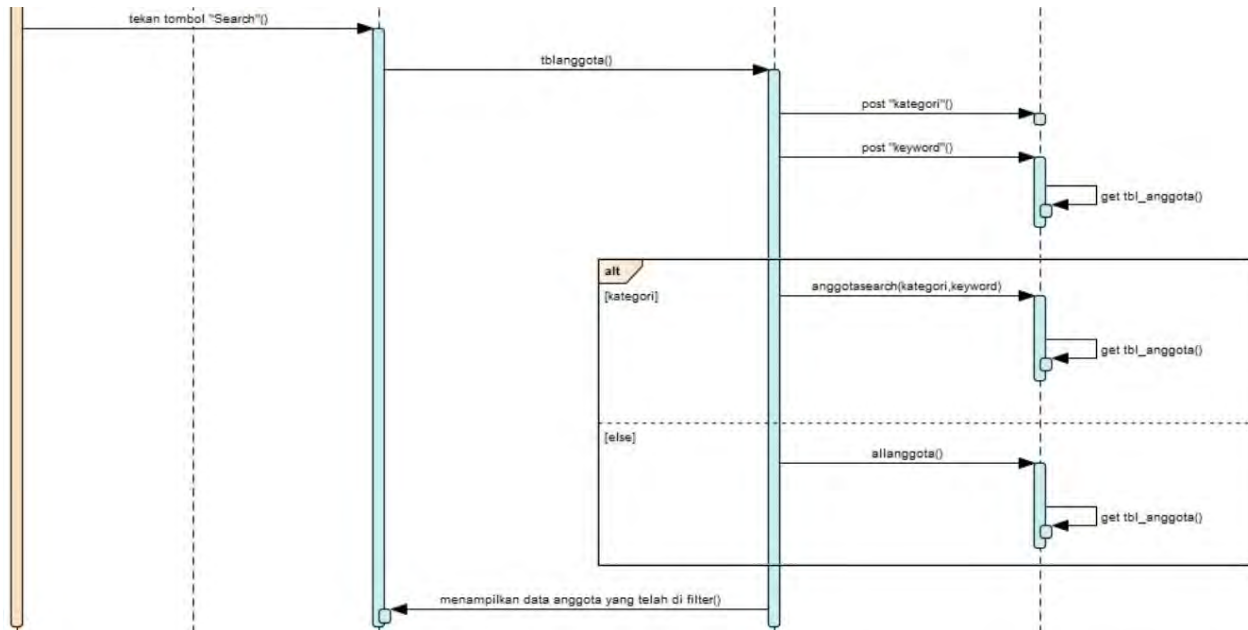
Gambar B.26 Display Data Anggota

*“Halaman ini sengaja dikosongkan”*



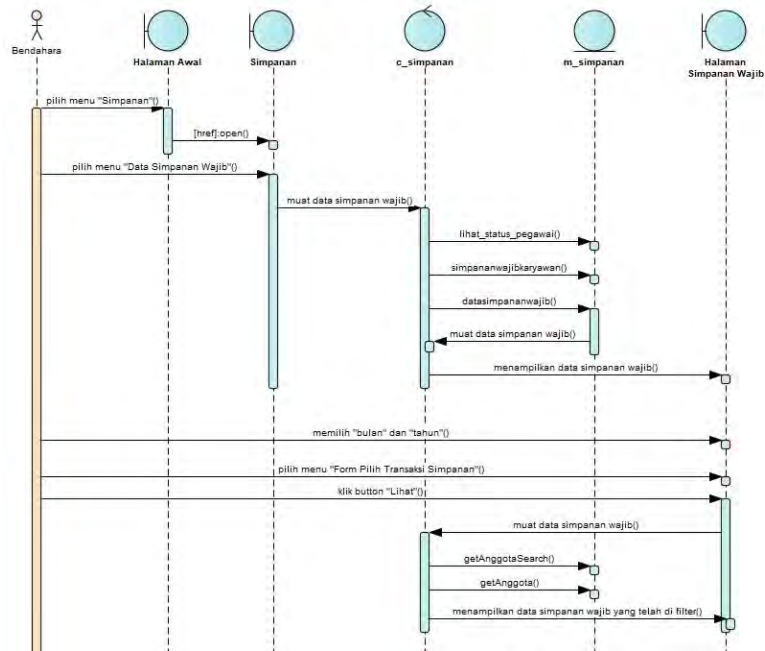
- Cari Data Anggota





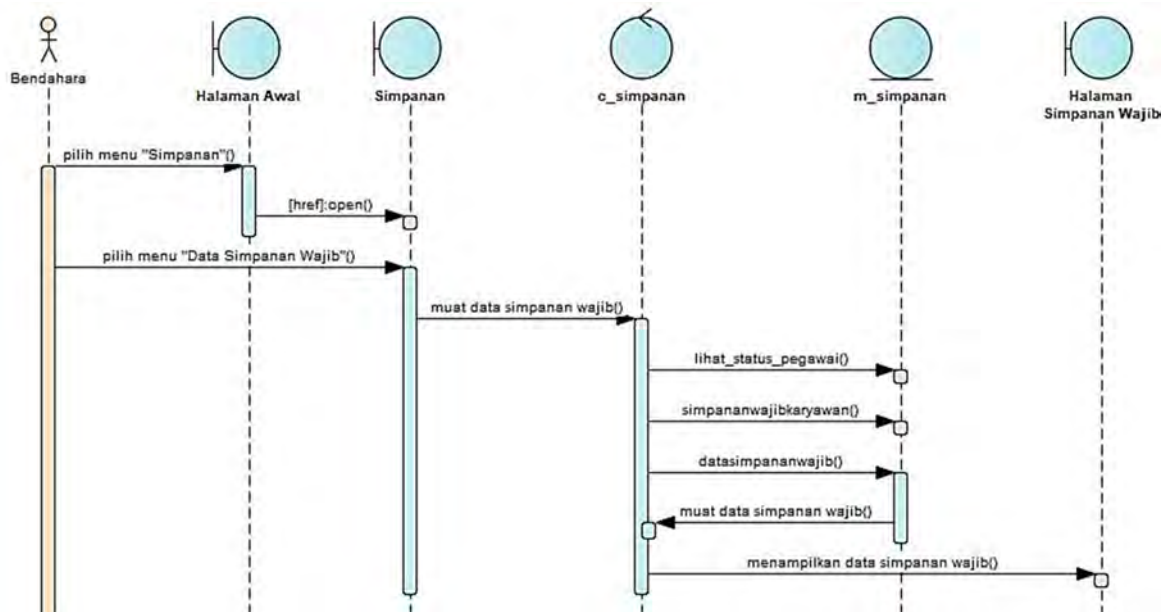
Gambar B.27 Cari Data Anggota

- Cari Data Simpanan Wajib**



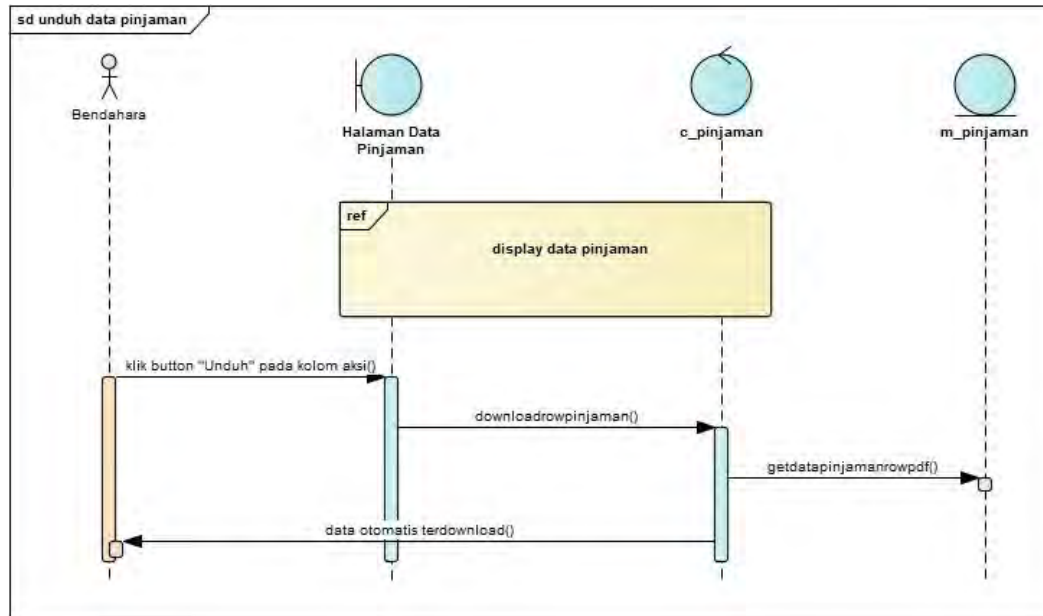
Gambar B.28 Cari Data Simpanan Wajib

- Display Data Simpanan Wajib



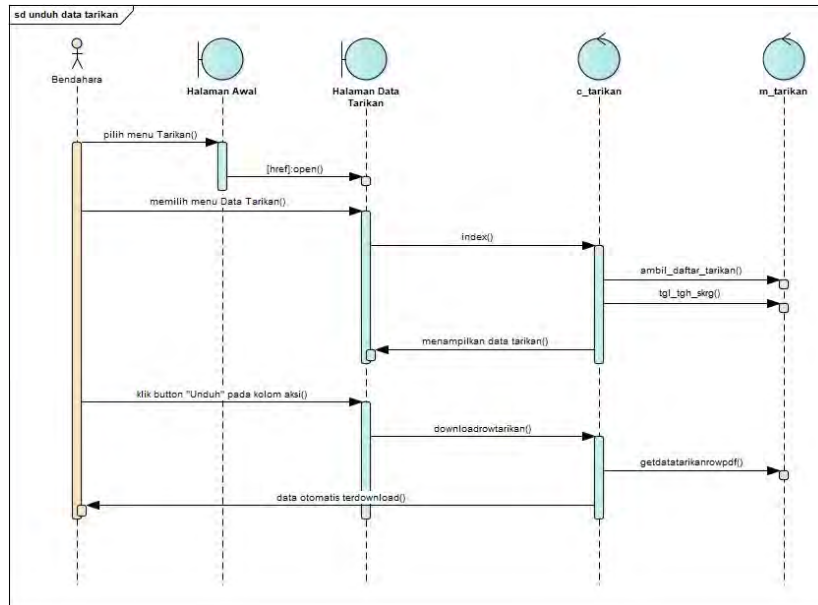
Gambar B.29 Display Data Simpanan Wajib

- Unduh Data Pinjaman



Gambar B.30 Unduh Data Pinjaman

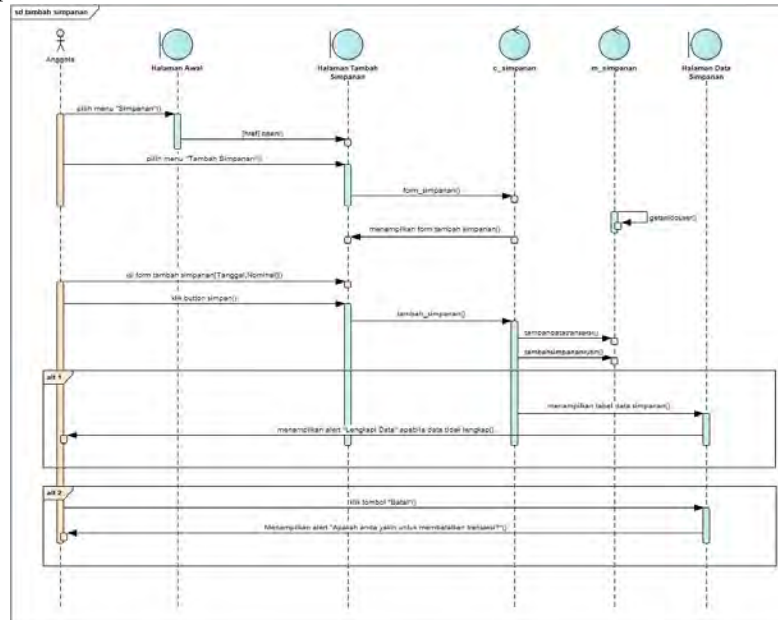
- Unduh Data Tarikan



Gambar B.31 Unduh Data Tarikan

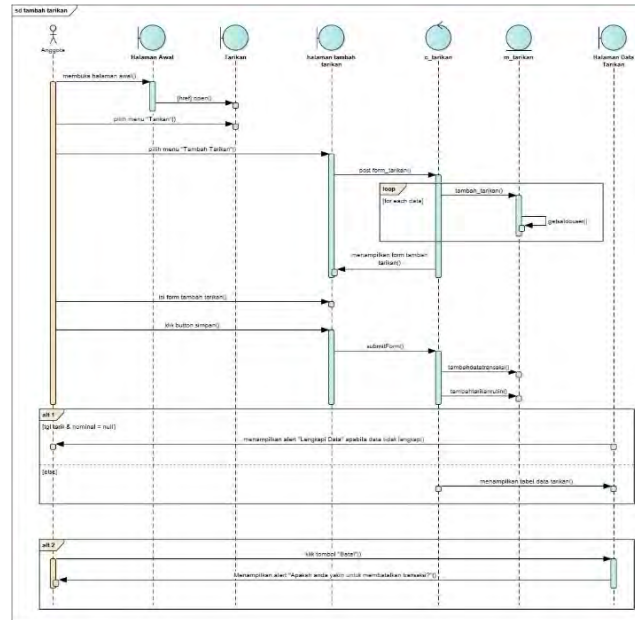
## Anggota

- Tambah Data Simpanan



Gambar B.32 Tambah Data Simpanan

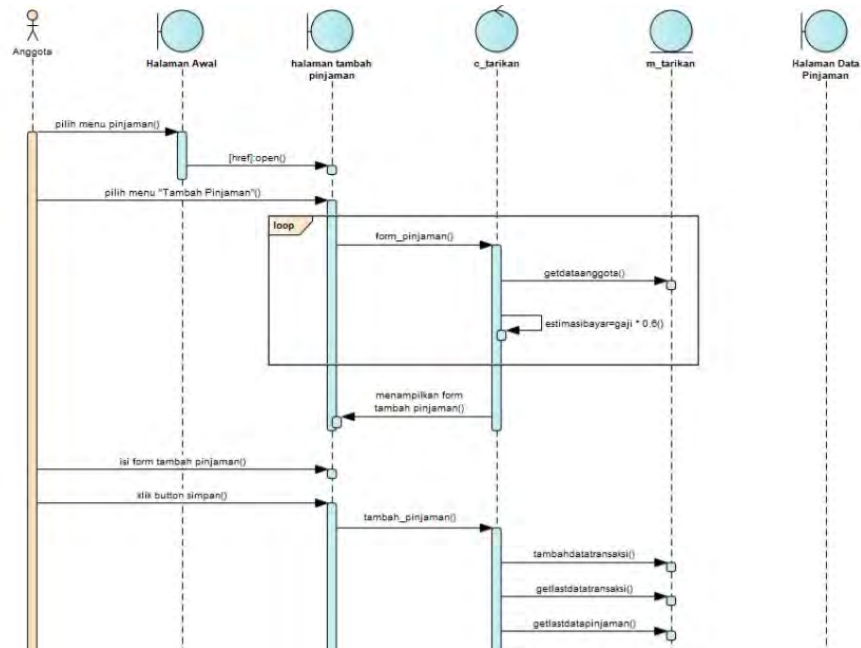
- Tambah Data Tarikan

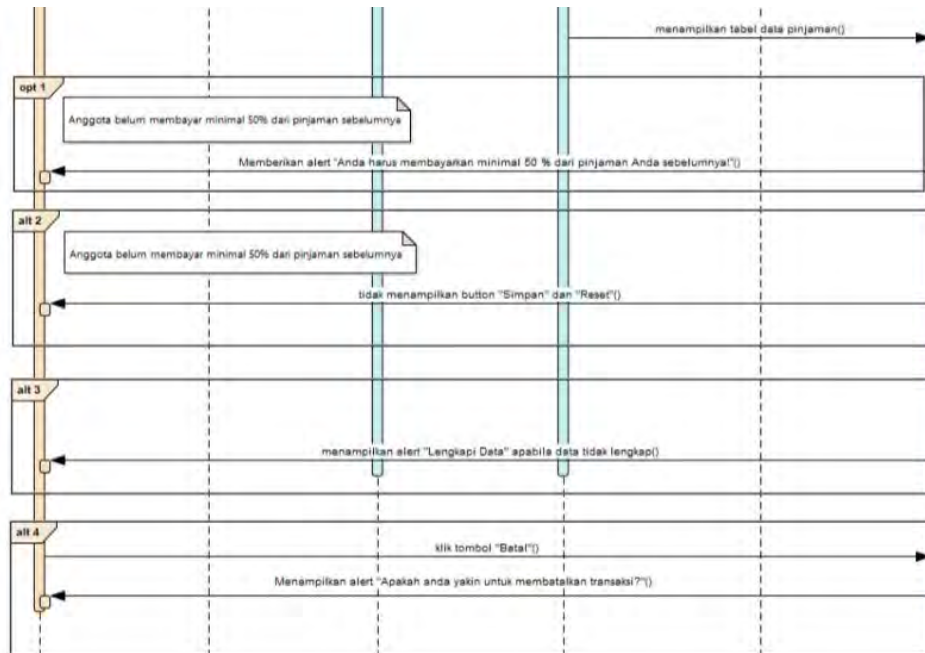


Gambar B.33 Tambah Data Tarikan



- Tambah Data Pinjaman



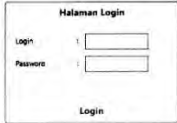





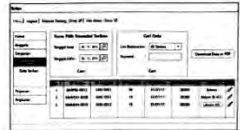




Gambar B.34 Tambah Data Pinjaman




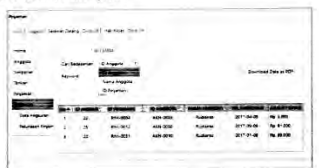


## C.2 DOKUMEN VALIDASI APLIKASI

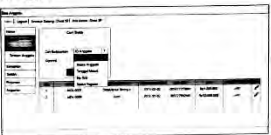
Nomor Proses	1.0
Input	Data Anggota
Nama Proses	Menampilkan Halaman Login
Deskripsi Proses	Pengguna (Anggota, Divisi SP, Ketua Koperasi, dan Bendahara Koperasi) dapat menggunakan fitur ini sebelum pengguna masuk ke halaman awal koperasi simpan pinjam
Output	<p>Halaman Login</p> 
Paraf	
Nomor Proses	1.1
Input	Data Anggota
Nama Proses	Menampilkan Data Simpanan
Deskripsi Proses	Divisi SP dapat menggunakan fitur display data simpanan untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	<p>Tabel Data Simpanan</p> 
Paraf	
Nomor Proses	1.2
Input	Data Anggota
Nama Proses	Menampilkan Data Tarikan


Nomor Proses	1.2
Input	Data Anggota
Deskripsi Proses	Divisi SP dapat menggunakan fitur display data tarikan untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	<p>Tabel Data Tarikan</p> 
Paraf	
Nomor Proses	1.3
Input	Data Anggota
Nama Proses	Menampilkan Data Pinjaman
Deskripsi Proses	Divisi SP dapat menggunakan fitur display data pinjaman untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	<p>Tabel Data Pinjaman</p> 
Paraf	
Nomor Proses	1.4
Input	Data Anggota
Nama Proses	Menampilkan Data Angsuran
Deskripsi Proses	Divisi SP dapat menggunakan fitur display data angsuran untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	<p>Tabel Data Angsuran</p> 

Paraf	<i>As</i>
Nomor Proses	2.1
Input	Data Anggota yang memiliki tunggakan pinjaman
Nama Proses	Menampilkan Form Tambah Daftar Tagihan
Deskripsi Proses	Divisi SP dapat menggunakan fitur tambah daftar tagihan yang digunakan untuk membuat daftar tagihan pelunasan pinjaman yang ditujukan untuk anggota.
Output	<p>Form Tambah Daftar Tagihan</p> 
Paraf	<i>As</i>

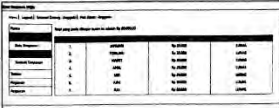
Nomor Proses	2.2
Input	Data Anggota yang telah melunasi pinjaman
Nama Proses	Menampilkan Data Pelunasan Pinjaman
Deskripsi Proses	Divisi SP dapat menggunakan fitur display data pelunasan pinjaman, yang didalamnya terdapat data-data anggota yang telah melunasi pinjaman.
Output	<p>Tabel Data Pelunasan Pinjaman</p> 
Paraf	<i>As</i>

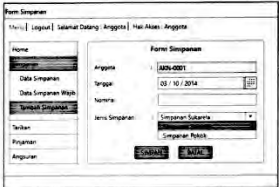
Nomor Proses	2.3
Input	Data Anggota yang telah melunasi pinjaman
Nama Proses	Tambah Anggota
Deskripsi Proses	Divisi SP dapat menggunakan fitur tambah data anggota. Didalam fitur ini terdapat sebuah form yang dimana Divisi SP harus mengisi seluruh data anggota kedalam form. Hasil dari form ini adalah username dan password yang dapat digunakan anggota untuk login kedalam aplikasi.

Nomor Proses	2.4
Input	Data Anggota
Nama Proses	Menampilkan Data Anggota
Deskripsi Proses	Divisi SP, Ketua, dan Bendahara dapat menggunakan fitur display data anggota yang terdaftar untuk melihat data-data anggota yang terdapat pada Koperasi Simpan Pinjam PT Wonosari Jaya
Output	<p>Tabel Data Anggota</p> 
Paraf	<i>As</i>

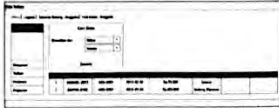
Nomor Proses	2.5
Input	Data Anggota
Nama Proses	Menampilkan Data Simpanan
Deskripsi Proses	Anggota dapat menggunakan fitur display data simpanan untuk melihat data-data transaksi yang telah tersimpan, serta dapat melihat apakah transaksi simpanan yang diinputkan telah di verifikasi atau belum, selain itu anggota juga dapat menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	<p>Tabel Data Simpanan</p> 
Paraf	<i>As</i>


Nomor Proses	2.6
Input	Data Anggota
Nama Proses	Menampilkan Data Simpanan Wajib
Deskripsi Proses	Anggota dapat menggunakan fitur display data simpanan untuk melihat data-simpanan wajib yang telah tersimpan sehingga memudahkan anggota untuk mengecek data simpanan wajib per bulan apakah telah dibayar atau belum. Selain itu anggota juga dapat menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.

Nomor Proses	2.6
Output	<p>Tabel Data Simpanan Wajib</p> 
Paraf	<i>[Signature]</i>


Nomor Proses	2.7
Input	Data Anggota
Nama Proses	Menampilkan Form Tambah Simpanan
Deskripsi Proses	Anggota dapat menggunakan fitur tambah simpanan. Didalam fitur ini terdapat sebuah form yang dimana Anggota harus mengisi seluruh data yang dibutuhkan kedalam form.
Output	<p>Form Tambah Simpanan</p> 
Paraf	<i>[Signature]</i>


Nomor Proses	2.8
Input	Data Anggota
Nama Proses	Menampilkan Data Tarikan
Deskripsi Proses	Anggota dapat menggunakan fitur display data transaksi untuk melihat data-data transaksi yang telah tersimpan, serta dapat melihat apakah transaksi tarikan yang didaftarkan telah di verifikasi atau belum, selain itu anggota juga dapat menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.


Nomor Proses	2.8
Output	<p>Tabel Data Tarikan</p> 
Paraf	<i>[Signature]</i>


Nomor Proses	2.9
Input	Data Anggota
Nama Proses	Menampilkan Form Tambah Tarikan
Deskripsi Proses	Anggota dapat menggunakan fitur tambah tarikan. Didalam fitur ini terdapat sebuah form yang dimana Anggota harus mengisi seluruh data yang dibutuhkan kedalam form.
Output	<p>Form Tambah Tarikan</p> 
Paraf	<i>[Signature]</i>

Nomor Proses	2.10
Input	Data Anggota
Nama Proses	Menampilkan Data Pinjaman
Deskripsi Proses	Anggota dapat menggunakan fitur display data transaksi untuk melihat data-data transaksi yang telah tersimpan, serta dapat melihat apakah transaksi pinjaman yang didaftarkan telah di verifikasi atau belum, selain itu anggota juga dapat menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.

Nomor Proses	2.10
Output	Tabel Data Pinjaman
Paraf	


Nomor Proses	2.11
Input	Data Anggota
Nama Proses	Menampilkan Form Tambah Pinjaman
Deskripsi Proses	Anggota dapat menggunakan fitur tambah pinjaman. Didalam fitur ini terdapat sebuah form yang dimana Anggota harus mengisi seluruh data yang dibutuhkan kedalam form. Apabila anggota belum membayarkan setidaknya 50% dari pinjaman sebelumnya, maka akan muncul notifikasi bahwa anggota harus membayarkan setidaknya 50% dari pinjaman sebelumnya.
Output	Tabel Tambah Pinjaman
Paraf	

Nomor Proses	2.12
Input	Data Anggota
Nama Proses	Menampilkan Data Angsuran
Deskripsi Proses	Anggota dapat menggunakan fitur display data transaksi untuk melihat data-data transaksi yang telah tersimpan, serta dapat melihat apakah transaksi angsuran yang diarahkan telah di verifikasi atau belum, selain itu anggota juga dapat menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	Tabel Data Angsuran
Paraf	


Nomor Proses	3.1
Input	Data Anggota
Nama Proses	Menampilkan Data Angsuran
Deskripsi Proses	Ketua dapat menggunakan fitur display data angsuran untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	Tabel Data Angsuran
Paraf	

Nomor Proses	3.2
Input	Data Anggota
Nama Proses	Menampilkan Data Pinjaman
Deskripsi Proses	Ketua dapat menggunakan fitur display data pinjaman untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	Tabel Data Pinjaman
Paraf	




Nomor Proses	3.3
Input	Data Anggota
Nama Proses	Menampilkan Data Tarikan
Deskripsi Proses	Ketua dapat menggunakan fitur display data tarikan untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	Tabel Data Tarikan 
Paraf	<i>[Signature]</i>


Nomor Proses	3.4
Input	Data Anggota
Nama Proses	Menampilkan Data Simpanan
Deskripsi Proses	Ketua dapat menggunakan fitur display data simpanan untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	Tabel Data Simpanan 
Paraf	<i>[Signature]</i>

Nomor Proses	4.1
Input	Data Anggota
Nama Proses	Menampilkan Data Simpanan
Deskripsi Proses	Bendahara dapat menggunakan fitur display data simpanan untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	Tabel Data Simpanan

Nomor Proses	4.3
Input	Data Anggota
Nama Proses	Menampilkan Data Angsuran
Deskripsi Proses	Bendahara dapat menggunakan fitur display data angsuran untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	Tabel Data Angsuran 
Paraf	<i>[Signature]</i>

Nomor Proses	4.5
Input	Data Anggota Pinjaman
Nama Proses	Menampilkan Data Pinjaman
Deskripsi Proses	Bendahara dapat menggunakan fitur display data pinjaman untuk melihat data-data transaksi yang telah tersimpan, dan menggunakan fitur cari data untuk memudahkan dalam pencarian data transaksi.
Output	Tabel Data Pinjaman 
Paraf	<i>[Signature]</i>

Gambar C.2 Dokumen Validasi Aplikasi

## C.3 ACCEPTANCE CHECKLIST

### LEMBAR PENGUJIAN FUNGSIONAL APLIKASI

Petunjuk pengisian :

1. Tulislah nama dan pekerjaan anda pada tempat yang telah disediakan
2. Berilah tanda centang (✓) pada kolom setuju atau silang (x) pada kolom tidak setuju

Nama : BAMBANG SRI WIBOWO, SE. MM.

Pekerjaan : KEMALA DIVISI SDM PUSAT KOPERASI KARYAWAN JAWA TIMUR

No.	Pertanyaan	Setuju	Tidak Setuju
1.	Aplikasi dapat membuat akun dengan baik	✓	
2.	Aplikasi dapat verifikasi dengan baik	✓	
3.	Aplikasi dapat login dengan baik	✓	
4.	Aplikasi dapat mengolah data dengan baik	✓	
5.	Aplikasi dapat mencari data dengan baik	✓	
6.	Aplikasi dapat melakukan verifikasi dengan baik	✓	
7.	Aplikasi dapat melihat informasi transaksi dengan baik	✓	
8.	Aplikasi dapat memudahkan Divisi SP dalam mengubah data	✓	
9.	Aplikasi dapat memudahkan Divisi SP dalam mengubah data	✓	
10.	Aplikasi dapat memudahkan anggota dalam pengecekan data transaksi	✓	
11.	Aplikasi dapat memudahkan anggota dalam menambahkan transaksi	✓	
12.	Aplikasi dapat memudahkan Bendahara dalam penagihan simpanan wajib	✓	

19/11/17  
/10  


Gambar C.3 Acceptance Checklist

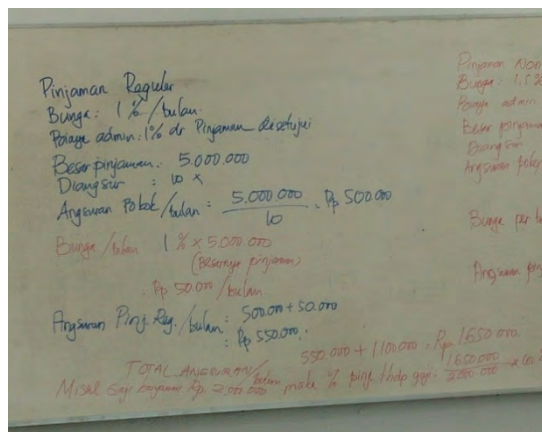
*“Halaman ini sengaja dikosongkan”*

## LAMPIRAN D

### Bukti Pendukung



Gambar D.1 Demo Aplikasi



Gambar D.2 Dokumentasi Koperasi

D - 2 -

*“Halaman ini sengaja dikosongkan”*